

100%  
Amiga

MovieSetter Reviewed

# Amazing COMPUTING™

Your Original AMIGA™ Monthly Resource

Vol. 4 No. 2 Feb. 1989  
US \$3.95 Canada \$4.95

## DATABASE REVIEWS:

Superbase PROFESSIONAL  
MICROFICHE Filer PLUS

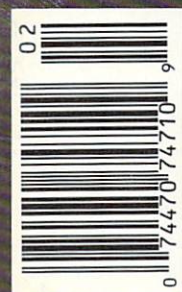
## PROGRAMMING:

Introduction to ARexx

## VIDEO:

Inside genlock

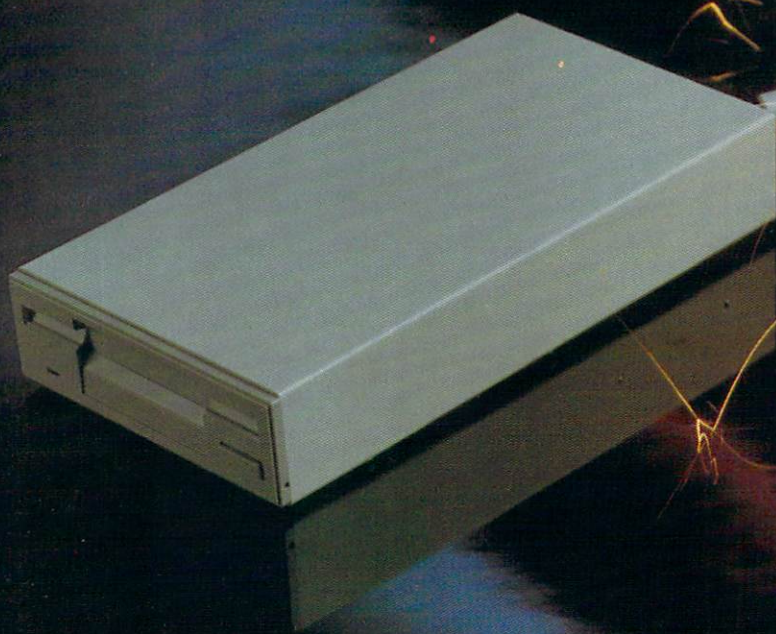
Plus Much More



Amazing Computing Your #1 Amiga Resource



# Our DRIVE is Dynamite!



## The AIR DRIVE

Dual-Sided, 880K

Color-coordinated chassis and faceplate  
Smaller and lighter than the Amiga drive

External drive pass-through

Extra long 30" interface cable

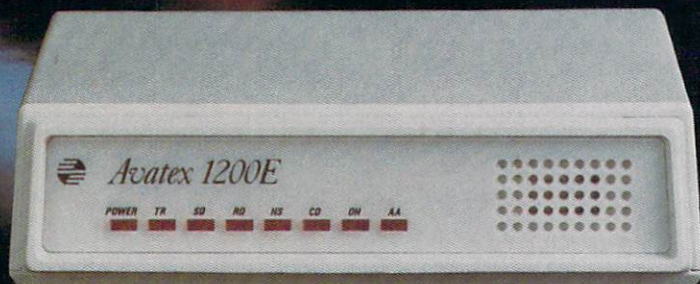
Spring-loaded dust door

Compatible with all Amiga systems

One year warranty

Internal drive also available

**CALL FOR DEALER NEAREST YOU.**



**\$79** Small and powerful.  
The Avatex 1200E.

## MEGATRONICS



To Order: 800-232-6342

Ask about our  
**LOWEST  
PRICE  
GUARANTEE**

INSIDE UTAH: (801) 752-2642  
FAX: (801) 752-8752  
CALL FOR A FREE CATALOG.  
CREDIT CARDS VERIFIED  
FOR YOUR PROTECTION.

MEGATRONICS, INC. BOX 3660, LOGAN, UT 84321

## Avatex MODEMS

\$105 .....	1200P
\$99 .....	1200HC
\$169 .....	2400
\$69 .....	1200i
\$159 .....	2400i

## FREE WITH EACH MODEM

Amiga communication software  
& CompuServe access time.



# TAX BREAK™

## Tax Preparation and Planning Tool

TAX BREAK™ is easy-to-use with on screen representation of forms and schedules, fast recalculation, and constant status of taxes due or refunds. Full integration of all forms and schedules allows constant updating of tax status for evaluation of different tax scenarios.

- Scrollable, on screen look-a-like forms and schedules
- Schedules can be opened from a menu or by double clicking on the schedule result line in form 1040 in US version or T1 for Canadian version
- Ongoing status of taxes due and refunds at bottom of each form, or expanded status screen available with mouse click or F2 key
- Instant updating of forms and schedules with each recalculation
- Line by Line data entry prompts
- On screen IRS booklet
- Prints IRS approved facsimile forms
- Pop up calculator
- Zoom in and out of interlace at will

US version 1.0 includes Form 1040, Schedules A-E, SE, 1906, 2441, IRA and more!

U.S. version 1.0  
\$79.95 Suggested Retail

contact:

**Oxxi inc.**

P.O. Box 90309  
Long Beach, CA 90809  
(213) 427-1227  
California Residents add 6.5%

Canadian version 3.0  
Known in Canada as Amiga-Tax™

\$69.95 Suggested Retail

contact:

**DATAMAX RESEARCH**

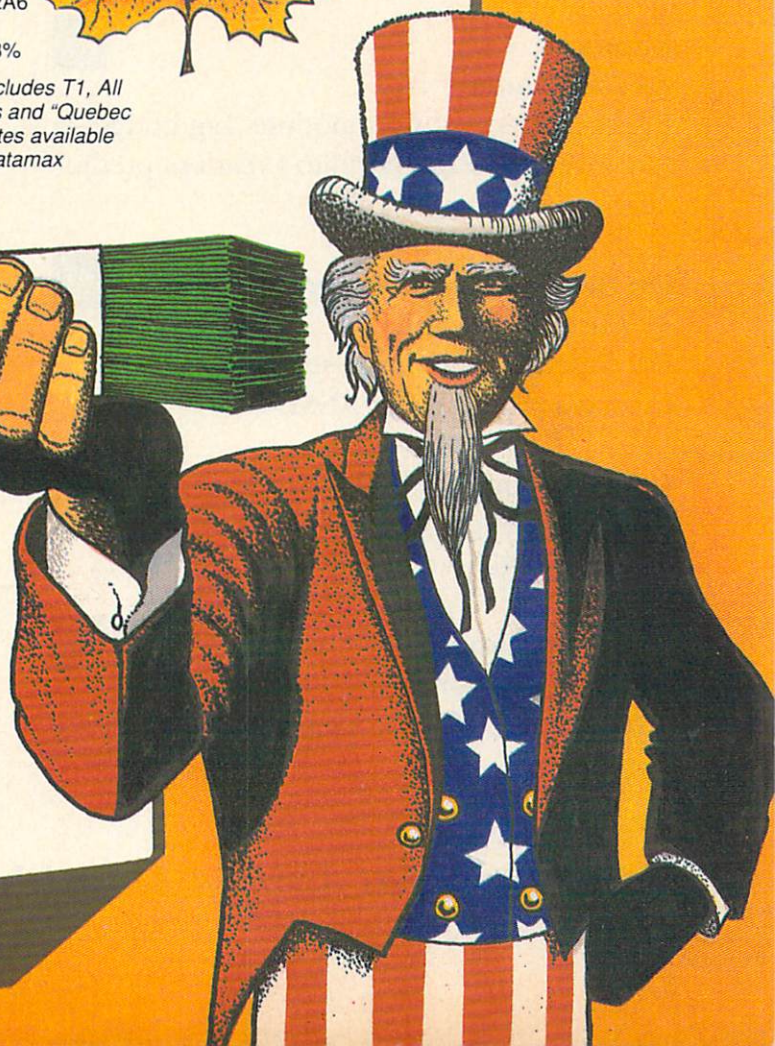
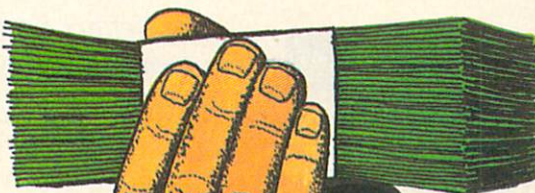
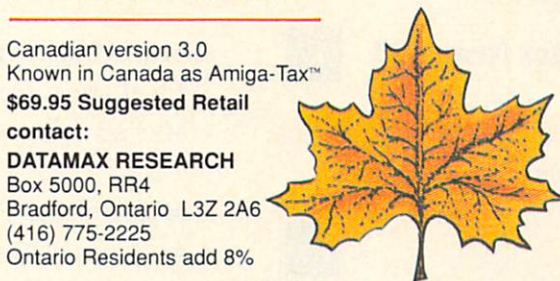
Box 5000, RR4  
Bradford, Ontario L3Z 2A6  
(416) 775-2225  
Ontario Residents add 8%

Canadian version 3.0 includes T1, All Schedules, all Provinces and "Quebec Forms" and more! Updates available for \$29.95 direct from Datamax Research

Dealers call: Southern Technology,  
American Software, Micro-Pace,  
Triangle or Air-Stat of Canada

See your local Amiga Dealer.

Amiga is a trademark of Commodore-Amiga, Inc.  
TAX BREAK is a trademark of Oxxi, Inc.  
Circle 75 on Reader Service card.





# Amazing COMPUTING™

## AMAZING COLUMNS

### Super Bug Bytes

by John Steiner

A double dose of bug repellant.

37

### New Products...and Other NeatStuff

by Michael Creeden

What's New? Deluxe Print II, The Talking Animator, 3 digit fun, and more.

47

### Roomers

by The Bandito

Big resignation at Commodore, big budget cuts at a major, and the Bandito's fearless predictions.

53

### On the Crafting of Programs

by David J. Hankins

Do we need a common standard for C programming? Look ANSI.

73

### C Notes from the C Group

by Steven Kemp

An introduction to unions

77

### The Command Line

by Rich Falconburg

A look at ED, your Workbench Screen Editor

87

## AMAZING REVIEWS

### MovieSetter: The Next Generation

by Steve Gillmor

A new level in Amiga animation.

15

### Superbase Professional

by Marion Deland

Marion takes a user's look at Superbase Pro.

23

### Microfiche Filer Plus

by Ronald Courrier

A multi-tasking database that uses ARexx to work with other programs

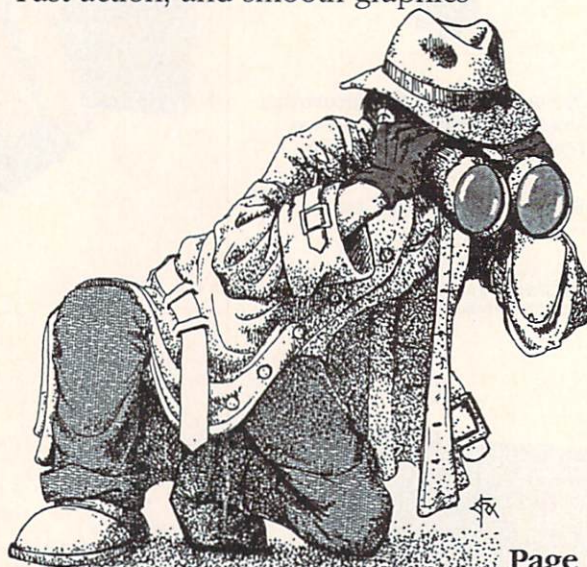
31

### Torch 2081

by Jeffery Scott Hall

Fast action, and smooth graphics

50



**Your Original AMIGA Monthly Resource**

Page 62



# • TABLE OF CONTENTS •

*Volume 4, Number 2*  
*February 1989*

## AMAZING FEATURES

### AmiForum

*by Steve Pietrowicz*

The Amiga pays a mid-winter visit to Mickey's Winter home for the first regional Amiga show.

35

### Max Morehead Interview

*by Richard Rae*

Rick talks to the creator of Moviesetter.

42

### SPY

*by Steve Faiwizewski*

Programming intrigue in Modula -2  
Interrogation techniques to learn your Amiga's vital information.

62

### Sync Tips

*by Oren Sands*

Inside the genlock; knowing what goes on "under the hood" could make your video better.

70

### A Common User Interface for the Amiga

*by Jim Bayless*

Does the Amiga need a kinder, gentler interface? Vote now!

79

### An Introduction to

#### ARexx programming

*by Steve Faiwizewski*

Climbing the Towers of Hanoi.  
Steve explains a hot programming environment for everyone!

91

### Crunchy Frog

*by Jim Fiore*

Take a look at this month's special Amiga-specific C programming.  
Please don't ask about the frog.

95

## AMAZING DEPARTMENTS

### From The Managing Editor

6

### Amazing Mail

8

### Index of Advertisers/ Reader Service Card

96

### Public Domain Software Catalog

105





# In this competitive world there is only one winner.

The results are in and Hurricane has won. According to the August AmigaWorld: "Currently, the Hurricane board is the best way to increase the performance of your Amiga."

Now Ronin has designed a whole family of Hurricane products that will make your Amiga 1000 or 2000 run up to 20 times faster. You'll save hours on ray tracing, desktop video, animation, and other math intensive applications.

The Hurricane family is fully modular, so you can start for as little as \$600 and upgrade in stages. Start racing with the 68020. You can then add a 68881. Later move up to our 68030 and maybe the 68882. Add up to 4 megabytes of 32-bit memory and your Amiga will sprint faster than a VAX minicomputer.

Why waste time and money on less performance. Run on in to your favorite dealer and get a Hurricane, or call us for more information. But whatever you do, don't be left at the starting line while everyone else passes you by.

**HURRICANE™** Dealer inquiries invited



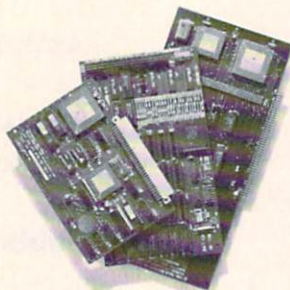
## Ronin

P.O. Box 1093, Alameda, CA 94501 U.S.A

**Telephone (415) 769-9325**

Fax (415) 865-9852 Twx 9102502720 Ronin UD

Amiga is a trademark of Commodore-Amiga, Inc. 68020, 68030, 68881, 68882 are trademarks of Motorola. Hurricane is a trademark of Ronin Research & Development Inc. VAX is a trademark of Digital Equipment Inc.



## Amazing COMPUTING™

### ADMINISTRATION

**Publisher:** Joyce Hicks  
**Assistant Publisher:** Robert J. Hicks  
**Circulation Manager:** Doris Gamble  
**Asst. Circulation:** Traci Desmarais  
**Asst. Circulation:** Donna Viveiros  
**Corporate Trainer:** Virginia Terry Hicks  
**Traffic Manager:** Robert Gamble  
**International Coordinator:** Marie A. Raymond  
**Marketing Manager:** Ernest P. Viveiros Sr.

### EDITORIAL

**Managing Editor:** Don Hicks  
**Editor:** Ernest P. Viveiros Jr.  
**Hardware Editor:** Ernest P. Viveiros Sr.  
**Music & Sound Editor:** Richard Rae  
**Senior Copy Editor:** Michael Creeden  
**Copy Editor:** Aimee Duarte  
**Copy Editor:** Elizabeth Fedorzyn  
**Copy Editor:** Keith Fox  
**Art Director:** William Fries  
**Photographer:** Paul Michael  
**Illustrator:** Brian Fox  
**Production Manager:** Donna M. Garant

### ADVERTISING SALES

**Advertising Manager:** Alicia Tondreau  
**Marketing Assistant:** Melissa J. Bernier

1-508-678-4200  
 FAX 1-508-675-6002

### SPECIAL THANKS TO:

Buddy Terrell & Byrd Press  
 Betsy Piper at Tech Plus  
 Bob at Riverside Art, Ltd.  
 Swansea One Hour Photo  
 PDQ Printing

Amazing Computing™ (ISSN 0886-9480) is published monthly by PiM Publications, Inc., Current Road, P.O. Box 869, Fall River, MA 02722-0869.

Subscriptions in the U.S., 12 issues for \$24.00; in Canada & Mexico surface, \$36.00; foreign surface for \$44.00.

Second-Class Postage paid at Fall River, MA 02722 and additional mailing offices.

**POSTMASTER:** Send address changes to PiM Publications Inc., P.O. Box 869, Fall River, MA 02722-0869. Printed in the U.S.A. Copyright©Nov. 1988 by PiM Publications, Inc. All rights reserved.

First Class or Air Mail rates available upon request. PiM Publications, Inc. maintains the right to refuse any advertising.

Pim Publications Inc. is not obligated to return unsolicited materials. All requested returns must be received with a Self Addressed Stamped Mailer.

Send article submissions in both manuscript and disk format to "Amazing Computing Submissions, C/O PiM Publications, Inc., P.O. Box 869, Fall River MA. 02722. Requests for Author's Guides should be directed to the address above.



# Amazing Dealers

The following are Amazing Dealers, dedicated to supporting the Commodore-Amiga™. They carry Amazing Computing™, your resource for information on the Amiga™. If you are not an Amazing Dealer, but would like to become one, call PiM Publications, Inc.:

1-508-678-4200

That Computer Place	Anchorage	Software City	Orange	Mayland	McP Associates	Ashtara	Maxwell Book Inc.	Desota
Juniata Electronics	Juneau	Software Kingdom	East Hartford	Barnd Treasures	Memorica	North Port	Metropolitan Computer	Richardson
Alabama		Tricom Computers	North Windham	Cal Com Inc.	Readers Market	Townandale	Micro Search	Houston
Abax Data Systems	Huntsville	Tricomp	Washington	CompuVision Vision Center	Software & Such	Software City	Software Library	Wichita Falls
Alabama Book Store	Tuscaloosa	Tricomp		Greetings and Readings	Software City	Software City	Software Terminal	Fort Worth
Command Computers	Birmingham	Tricomp		Knowledge Systems	Software City	Software City	The Computer Experience	San Antonio
Madison Book & Computer	Madison	Tricomp		Micro Computer Center	Software Supermarket	Software City	The Computer Shop	Amarillo
M&H Photo & Computer Shop	Montgomery	Tricomp		Professional Micro Services	Star Tech Systems	Software City	The Federated Group	Mosquito
The Computer Niz	Birmingham	Tricomp		Software Advantage	Synapse Trading	Software City	UAC	Salt Lake City
The Computer Image	Birmingham	Tricomp		Software & Things	Synapse Trading	Software City	Magazines and Gifts	Provo
Universal Computer System	Mobile	Tricomp		Software & Things	Synapse Trading	Software City	People's Computers	
Adams		Tricomp		Software & Things	Synapse Trading	Software City	Vermont	South Burlington
Commodore Connection	Shenwood	Tricomp		Software & Things	Synapse Trading	Software City	Relay Supply Inc.	
Minotrix	Fort Smith	Tricomp		Software & Things	Synapse Trading	Software City	Johnson City	
The Micro Shop	Little Rock	Tricomp		Software & Things	Synapse Trading	Software City	Brooklyn	Fairfax
Arizona		Tricomp		Software & Things	Synapse Trading	Software City	Family Video & Computer Center	Newport News
Books Brothers Ltd.	Tucson	Tricomp		Software & Things	Synapse Trading	Software City	JO Computer	Norfolk
Computer Library, Inc.	Tempe	Tricomp		Software & Things	Synapse Trading	Software City	World Computers Inc.	Richmond
Computer Business Systems	Tempe	Tricomp		Software & Things	Synapse Trading	Software City	Tract Software	Prince William
North American Digital	Phoenix	Tricomp		Software & Things	Synapse Trading	Software City	Software Systems	Woodbridge
Try Tech Inc.	Phoenix	Tricomp		Software & Things	Synapse Trading	Software City	Washington	
California		Tricomp		Software & Things	Synapse Trading	Software City	Amazing Electronics	Spokane
3 E Software And Systems	Hayward	Tricomp		Software & Things	Synapse Trading	Software City	A.P.P.E. CO-OP	Renton
American Mags & Books Ltd.	Emeryville	Tricomp		Software & Things	Synapse Trading	Software City	Apple Valley Computer Center	E. Wenatchee
August	Fremont	Tricomp		Software & Things	Synapse Trading	Software City	Bit Bits and Nobles	Spokane
Balboa Office Supply	San Diego	Tricomp		Software & Things	Synapse Trading	Software City	Bull Dog News	Seattle
B&B Bytes	Fairfield	Tricomp		Software & Things	Synapse Trading	Software City	Byte World	Kent
B&B Paces Comp Prod.	San Diego	Tricomp		Software & Things	Synapse Trading	Software City	Computer Exchange Services	Bellingham
Boldland	San Diego	Tricomp		Software & Things	Synapse Trading	Software City	Computer Data	Seattle
Brown Knives Computing	Redlands	Tricomp		Software & Things	Synapse Trading	Software City	Computers + Inc.	Tacoma
CAL-Tech Bookstore	Pasadena	Tricomp		Software & Things	Synapse Trading	Software City	Family Computers	Everett
Candy Computer	Elk Grove	Tricomp		Software & Things	Synapse Trading	Software City	Micro Age Computer Store	Bellevue
Cardinal News	Costa Mesa	Tricomp		Software & Things	Synapse Trading	Software City	Nybbles & Bytes	Tacoma
Central Park Bookstore	San Mateo	Tricomp		Software & Things	Synapse Trading	Software City	On International	Seattle
Century Computer Systems	La Habra	Tricomp		Software & Things	Synapse Trading	Software City	Port Book and News	Port Angeles
Circus of Books	Los Angeles	Tricomp		Software & Things	Synapse Trading	Software City	Programs Plus	Seattle
Circus of Books	Los Angeles	Tricomp		Software & Things	Synapse Trading	Software City	Real Time Systems	Seattle
Commodore Land	Alhambra	Tricomp		Software & Things	Synapse Trading	Software City	Take A Byte	Kennecook
Computer Attic	National City	Tricomp		Software & Things	Synapse Trading	Software City	The Computer Shop	Yacouver
Computer Library	Palo Alto	Tricomp		Software & Things	Synapse Trading	Software City	Tower Books	Bellevue
Computer Library Books	San Jose	Tricomp		Software & Things	Synapse Trading	Software City	Y Not Magazines and Books	Federal Way
Computer Nook, Inc.	San Jose	Tricomp		Software & Things	Synapse Trading	Software City	Waco	Waco
Computer Outlet	San Diego	Tricomp		Software & Things	Synapse Trading	Software City	20th Century Books	Madison
Computer Out	San Jose	Tricomp		Software & Things	Synapse Trading	Software City	Computer Software Center	Milwaukee
Computer Spectrum	Burlingame	Tricomp		Software & Things	Synapse Trading	Software City	Fox Valley Personal Computer	Ophosh
Computer Time	Clive Hts	Tricomp		Software & Things	Synapse Trading	Software City	Madson Programmer's Guide	Madison
Concord Newstand	Concord	Tricomp		Software & Things	Synapse Trading	Software City	Mayfair Computers	Milwaukee
Covina Bookstore	Covina	Tricomp		Software & Things	Synapse Trading	Software City	Mon's Computers	Severn Park
Dataphile	Watsonville	Tricomp		Software & Things	Synapse Trading	Software City	North City Software	La Crosse
Delaware's News Agency	Oakland	Tricomp		Software & Things	Synapse Trading	Software City	Software City	Wausau
Discount Computer Station Inc.	San Jose	Tricomp		Software & Things	Synapse Trading	Software City	TMW Software Inc.	Waukegan
Empire Computers	San Jose	Tricomp		Software & Things	Synapse Trading	Software City	Waco	Waco
Eremitas News	San Jose	Tricomp		Software & Things	Synapse Trading	Software City	Waco	Waco
EOX Computer Co.	Walnut Creek	Tricomp		Software & Things	Synapse Trading	Software City	Waco	Waco
Fairchild Computers + Software	Rancho Cucamonga	Tricomp		Software & Things	Synapse Trading	Software City	Waco	Waco
Hedding Way	Stockton	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Heinrichs, Schneider & Young, Inc.	Berkeley	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	Concord	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	Home Computing Center	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	San Leandro	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	San Leandro	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	San Leandro	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	San Leandro	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	San Leandro	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	San Leandro	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	San Leandro	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	San Leandro	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	San Leandro	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	San Leandro	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	San Leandro	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	San Leandro	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	San Leandro	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	San Leandro	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	San Leandro	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	San Leandro	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	San Leandro	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	San Leandro	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	San Leandro	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	San Leandro	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	San Leandro	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	San Leandro	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	San Leandro	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	San Leandro	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	San Leandro	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	San Leandro	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	San Leandro	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	San Leandro	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	San Leandro	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	San Leandro	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	San Leandro	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	San Leandro	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	San Leandro	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	San Leandro	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	San Leandro	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	San Leandro	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	San Leandro	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	San Leandro	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	San Leandro	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	San Leandro	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	San Leandro	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	San Leandro	Tricomp		Software & Things	Synapse Trading	Software City	Waco	
Home Computing Center	San Leandro	Tricomp		Software & Things	Synapse Trading</			

Amazing Computing™ is also available in most B. Dalton Booksellers, B. Dalton Software Stores, Crown Books, Software Etc., selected WaldenBooks Stores, and Walden's Software Store locations



# From The Managing Editor:

## Three Years!

That's right. This issue of *Amazing Computing™* celebrates our three year anniversary. We are filled with a great deal of awe and pride as we prepare this issue.

Our awe is based on the rapid growth and diversity of the Amiga. The Amiga is being utilized in more areas by more people than ever before. With the Amiga, a computer user can prepare a document, produce a presentation, create fascinating three-dimensional graphics, control machinery, teach, learn, explore, and even play a game.

Our pride is in the consumer acceptance of the Amiga. At press time, I have been told Commodore has announced over one million Amiga computers sold worldwide. One million computers represents a fantastic growth from the 600,000 units of a few months ago. This accelerated growth can only mean the general consumer is now aware of what we have known all along: The Amiga is a powerful, reasonably priced computer which will take its users into areas no other machine can match.

While other computers touted their ability to produce written documents and page layouts, Amiga supplied these documents in color. In fact, the Amiga's main problem has been the inability of third party hardware suppliers to deliver the same price and performance ratio of a color output which we have been treated to with the Amiga. Although brilliant color reproduction is available, it comes at a price the average user cannot afford.

However, the Amiga's importance is not in documentation, but in presentation. Amiga has redefined the term Desktop Presentation. Amiga not only makes desktop presentation affordable through economic video and graphic software and hardware, but the Amiga was the first machine to make these techniques available to the average user.

## Genlock and The Amiga

I still remember the Commodore booth at Comdex in Atlanta in 1986. An Amiga was attached to a video camera in a small booth in one area. There were no salespeople, no Commodore personnel were manning the area. The Amiga stood silently, its video camera pointed at the front of the booth. As you passed in front of the Amiga, you passed into the camera's field of vision. When you looked at the Amiga monitor, you were greeted by Amiga-generated fonts telling you about Amiga Genlock. Behind the fonts, you could see your picture from the video camera in live action on the Amiga screen.

Commodore has announced over one million Amiga computers sold worldwide. One million computers represents a fantastic growth from the 600,000 units of a few months ago.

It was a while before this technology was available to the Amiga user, yet it was two years before I saw the same demonstration performed on another computer. I watched as the crowd surrounded the demonstration. Each person oohed and ahed over this new innovation. The price for this computer with its equipment was enough to buy two Amiga computers equipped with Genlocks and supporting software.

I wanted to grab each person and tell them this technology had been available for years on a computer that just happened to also be multitasking. As I watched the crowd move through the booth, I knew there was little chance of reaching them. They were reacting instead of thinking, and price was no object.

Still, price is of no consequence if quality is lacking. There are many things I would change on the Amiga if I had the power. However, I am not certain my changes would improve the overall machine for everyone. My changes are things I would do for my use based on my needs. Except in the Amiga, such customization is possible and even encouraged.

In short, the Amiga was introduced three and a half years ago and although new versions of the Amiga have been introduced and accepted by the public, the main graphics and sound capabilities have remained unchanged. Yet this technology is beyond the scope of most of the computers available today. Those that do mimic the Amiga's ability do so at a cost far beyond the Amiga's.

## Amazing Computing™ & Amiga

I began this editorial by talking about the awe and pride we feel as we produce this issue. These emotions extend to our own publication and your acceptance of it. We look with both awe and pride at what we have been able to accomplish in the past three years.

With the help of a great many Amiga enthusiasts, we have been able to expand the "pool of knowledge" of the Amiga Computer beyond our greatest expectations. Through hardware and software submissions by our readers, we have been able to demonstrate the abilities of the Amiga both great and small. These applications were operations which could easily be performed by our readers. OK, maybe they were not all easily performed by our readers, but they were available.

Amazing Computing™ has been allowed to pace its growth with that of the Amiga. Our first issues were the founding of a commitment to the Amiga user to continually improve our ability to inform. We have added color, produced more hardware articles and spent our time creating an environment our readers enjoy. This is important to us.

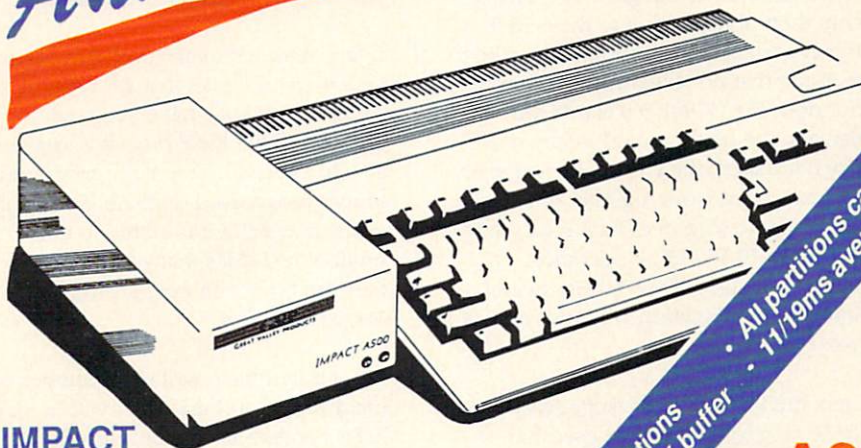
*Continued on page 12*



# AMIGA 500 USERS! NOW GET A2000 POWER!!

*Announcing*

THE WORLD'S FIRST  
AUTOBOOTING HARD DISK  
FOR THE AMIGA 500



## IMPACT A500 – HD/RAM SUBSYSTEM

offers everything any A500 owner has ever dreamed of – in one compact, easy-to-install, add-on subsystem; provides the same high performance as our top-of-the-line A2000 controller

IMPACT™ A500 – HD/20  
A500 – HD/30  
A500 – HD/40Q  
A500 – HD/45  
A500 – HD/80Q

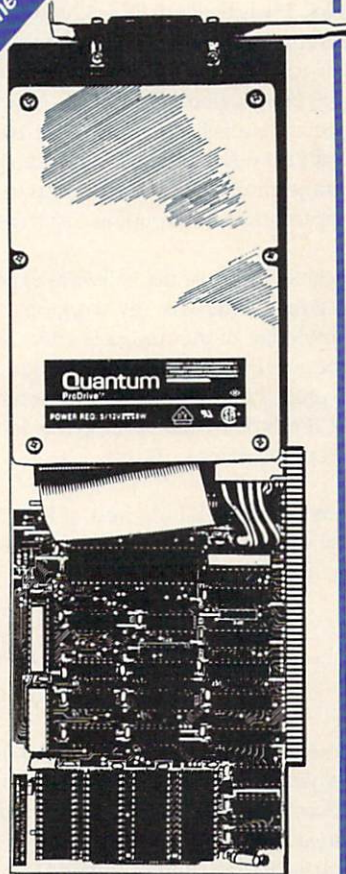
Available with a built-in 20, 30, 40, 45 or 80 Mega Byte Hard Drive and an optional 2MB plug-in FAST RAM/AUTOBOOT expansion module.

## A2000 USERS!

*Still*

THE MOST  
POWERFUL HARD DISK  
SOLUTION FOR  
YOUR A2000!

IMPACT™ A2000 – Hard Card 30  
A2000 – Hard Card 45  
A2000 – Hard Card 40Q  
A2000 – Hard Card 80Q  
A2000 – Hard Card 100



The GVP IMPACT-HC series of "Hard-Cards" for the Amiga 2000 matched with the 3.5" Quantum ProDrive hard disk drives, provide turbo performance and unbeatable value at list prices as low as \$18.00 per Mega Byte! The Quantum ProDrives provide average access times as low as 11ms with their built-in 64KB read-ahead cache and offer a MTBF of 50,000 hours which is almost twice the industry average.

*Now  
Shipping*

## AUTOBOOT EPROMS



GREAT VALLEY PRODUCTS INC.

PLANK AVENUE, PAOLI, PA 19301 TEL (215) 889-9411 FAX (215) 889-9416



## A WINNER!!!!

### Dear Amazing Computing,

I would like to thank everyone at Amazing Computing™, for the wonderful gift collection of public domain software.

I admit, at first I was a little disappointed. My experience with public domain software has been very limited. But once I began to sample a few of the disks, I was thrilled. Thank you for sending both the AMICUS and the Fred Fish collections. Either collection would have been a fine gift in itself. And a special thanks for including the 10 latest Fred Fish disks and, also, the handsome storage cases. The collection came in time to make the holidays just super.

Because much of the software explores the Amiga internals, my working knowledge of the Amiga is growing stronger. Every facet of the Amiga is covered. I'm even testing out one of the ASCII editors, while writing this letter. And those demos...Wow!

Here's wishing all the best of luck in the new year. And by the way, cast another vote to keep the Bandito.

Sincerely,  
Ron Wilson  
Santa Ana, CA.

*We are extremely pleased that you are enjoying the prize. We have always believed the best place to start using your Amiga is through the continually expanding selection of Public Domain Software. It is one way an Amiga user can experience a variety of directions for the Amiga without spending a fortune. Thank you for your note.*

### A Few Words on GOMF

#### Dear AC:

While I was, of course, pleased to see our product, GOMF 3.0, mentioned in your August issue article, "Video in the Sunshine State", there were some inaccuracies which were presented.

The article states that, 'The hardware device is installed under the AGNUS chip...' This is NOT the case, and must have caused considerable consternation

in both Amiga 1000 (Thin Agnus) and 500/2000 (Fat Agnus) owners. The fact is, the GOMF 3.0 Hardware option plugs into the PAULA socket (The PAULA chip is the same in all Amigas). The PAULA chip then plugs back into the GOMF Hardware socket. We would also like to mention that no soldering is required, nor does the GOMF 3.0 Hardware damage the motherboard in any way. Once installed, the GOMF 3.0 software and hardware work together to actually allow you to skip over fatal errors and SAVE YOUR WORK. Under most conditions, the GOMF 3.0 Hardware/Software combination will even unlock a frozen machine.

I feel the features are substantially more interesting to the average user than reading that "when an error occurs you simply press the button and information about the error is dumped." This IS true, but somehow seems to imply that GOMF is only an esoteric debugging tool for advanced programmers. Much more importantly, the article neglected to mention that after GOMF traps and eliminates the error, the program AND data are still usable. I suspect that 'average users' will appreciate the cure more than the diagnosis, so to speak.

Again, we realize that the article was not designated as a review of GOMF, but we would like to set the record straight. Thanks for a wonderful magazine. Amaze long and prosper!

Yours Truly,  
Graeme Bennett  
Hypertek/Silicon Springs

*Thank you for bringing the error of the chip placement to our attention. As you stated, the article was not a review of GOMF, but an overview of the work RGB Video Creations is doing to advance Amiga use in video. Mr. Pietrowicz covered 24 products and/or projects in one and three quarter pages with two photos. Thankfully, we paid Steve by the page, and not by the products covered.*

### Dear AC:

I have been a subscriber since your fourth issue and intend to keep on subscribing, so you folks must be doing something right. Maybe the following will help some of your readers.

I have read with interest about various criticisms and bugs that show up in Amiga Basic but I have yet to see mentioned the ELSE bug (this applies also to ELSE IF). This bug shows up in longer programs. I have not been able to determine if there is a limit to the number of ELSE's that can be used or if they are limited in longer programs. Maybe it is both.

I do know I have written a number of programs of 40K and over with a lot of ELSE's and in all of them the ELSE bug bites. When you try to run such a program in Amiga Basic you will get Syntax Error message, the program aborts, the program segment is listed, and the editor will outline a box in the cursor color immediately after the closing END IF of the offending structure. This box is of various sizes, anywhere from a one character size up to 4 or 5 characters. There seems to be no reason behind the box size. This really can be puzzling the first few times it happens, because there is not an error, just a bug. However, you must go back and change the routine so no ELSE is used.

It is interesting that I can leave the ELSE's in and compile and run a program satisfactorily using AC/Basic. By the way, I have tried Amiga Basic from 3 different original disks, the 1.1 version, the 1.2 version I bought for my 1000, and a 1.2 version that came with my 2000, and they all have the ELSE bug.

While I am on the subject of Amiga Basic, I timed the loading of a 74k program using the Amiga Basic editor. It took 1 minute 10 secs to load. I then timed the loading of the same program with Aedit, a commercial editor (I am using it right now) and it took only 18 secs. You can really save time while developing programs by using a good editor.



# WIN A MONITERM!

Build an expert system application with the MAGELLAN™ artificial intelligence expert system building tool and win a MONITERM Viking I high resolution monitor for your AMIGA.

Imagine the possibilities: the power of the AMIGA coupled with the flexibility of MAGELLAN!

Categories for the First Annual MAGELLAN Application Contest are:

**Most Productive Application**

Business or Diagnostic systems, decision assistance tools for financial investment, production control systems, or inventory management systems.

**Most Creative Application.**

Music or Animation systems, or innovative approaches to traditional expert systems.

**Most Bizarre Application.**

Let your imagination run wild!

**3 Grand Prize Winners** will receive:

A Moniterm Monitor

A MAGELLAN Developer Package

Retail Value \$3000.

**2 Runners-Up** in each category will receive:

MAGELLAN Developer Package

Retail Value \$1000.

All finalists will also receive a MAGELLAN designer sweatshirt.

All entries must be received by Emerald Intelligence by June 1, 1989. Contest void where prohibited. Contest Winners will be announced at Ami-Expo Chicago in July 1989. All entries become the exclusive property of Emerald Intelligence Inc., and cannot be returned. Winning entries may be used in a promotional campaign and any entry may be included in promotional materials or application manuals. For a complete set of rules and an entry form, send a self addressed stamped envelope to MAGELLAN Contest Rules, c/o Emerald Intelligence.

# MAGELLAN™

THE EXPERT SYSTEM SOFTWARE

Emerald  
Intelligence

334 South State Street, Ann Arbor, Michigan 48104. (313) 663-8757

MAGELLAN is a trademark of Emerald Intelligence Inc. AMIGA is a registered trademark of Commodore-Amiga Inc. Moniterm is a trademark of Moniterm Corporation.



Microsoft should be ashamed of itself for publishing such a poor, slow editor and leaving the bugs in Amiga Basic for so many years. It certainly does not give you much confidence in them.

Regards,  
C.A. Barringer  
Crescent City, FL.

*We appreciate your letter, however, we do not have an immediate answer. Unfortunately, you did not send a copy of your suspected code and we were unable to reconstruct your problem. If any other readers can verify what is happening, please drop us a note.*

## AMIGA USER GROUPS!

### Dear AC,

This is to inform you that we are organized as the AmAz! (pronounced Amaze) Amiga User Group of Arizona. Our group name stands for Amiga Arizona, and I believe we are the largest organized Amiga User Group in the state. Our regular monthly meeting membership, which is held on the third Thursday evening of the month, is well over one hundred (100) strong, with a paid membership of one-hundred thirty-five (135), and growing monthly.

We meet in an auditorium at the Glendale Public Library here in Glendale, Suburb of Phoenix. Our presentations include a color 25" monitor and a high quality sound system with microphone and stage presentation. We also own our own Amiga 2000 computer and operate our own BBS (300/1200/2400) at 602-843-6574.

We also encourage any product presentations. Such a presentation may be scheduled in advance. If anyone is interested in presenting a product or would like to send a demo of a product, please contact us at AmAz! either on our BBS, or write to us at the following address:

Amiga Arizona (AmAz!)  
P.O.Box 82371  
Phoenix, AZ 85071-2371

Yours Truly,  
Stan Krawczyk,  
President

### Dear Amazing Computing:

The Amiga Users' Group of South Jersey has been going strong for over two years. We meet on the 3rd Tuesday of each month at 7:30 PM (except during the summer) at American Legion Post #38, 429 Hawthorne St., Haddonfield, NJ. We publish a newsletter, AmigaDEX, and have a Public Domain exchange.

Our address is:  
P.O.Box 3761  
Cherry Hill, NJ 08034  
(609) 667-2526

Thank you for your support of Amiga user groups.

Sincerely,  
Samuel S. Johnson  
Publicity Chairman

### Dear AC:

We would appreciate it if you would include the Port Orange Commodore User Group (POCUG) in your user group list. We meet each Saturday at the Darrel Kreighbaum Memorial Library in Port Orange, Florida, from 2:00 to 5:00 pm.

Our mailing address is:  
POCUG  
1244 Thomasina Drive  
Port Orange, FL 32019-7437  
Current club officers are:  
President/Librarian- Rick Stidham  
VP/Secretary-Steve Gauthier  
Treasurer-Marjorie Halloway

Our BBS is Tec-Net (SysOps: Dr. Shade, Micro, and WWooly), which is online 24 hours a day, 7 days a week, at 300/1200 baud, and has an Amiga message base and an Amiga upload/download area. POCUG can also be reached by voice at (904) 767-0545 BEFORE 9:00 pm EST.

We have started a free trading post service for Amiga HAM and IFF pictures. Just send a 3.5" disk with some of your favorite picture files, and return postage, and we'll fill your disk with picture files that others have sent and return it to you. Feel free to send as many or as few files as you like (though we appreciate as many as you can send), or even multifile disks, members and non-members.

Thank you for your help!

Best regards,  
Rick Stidham  
President,  
Port Orange  
Commodore User Group

### Dear Amazing Computing,

First of all, let me introduce myself. I'm the librarian for our local User Group, and I'd appreciate it if you would add the following to your User Group list:

Ames Area Amiga Users Group  
Welch Avenue Station  
P.O. Box 1011  
Ames, Iowa 50010

Our club officers are:

President—Scott Fredericksen  
Vice-President—Jeff Miller  
Vice-President—Red Varnum  
Secretary/Treasurer—Brian Hausauer  
Newsletter Editor—Lynn Ritter  
Librarian—Mike Druess

We have approximately 40 members in our group, many of whom are faculty or students of Iowa State University, or members of the surrounding communities. In addition, we publish a monthly newsletter called 'Amiga Output' and a monthly club disk. We have over 275 disks in our public domain library, the descriptions of which are on our Library On A Disk (L.O.A.D.) disk. Our group started in September, 1986, and we just celebrated the 2nd anniversary of our club.

We meet monthly in the basement of Midland Financial Saving and Loan, 525 Main Street, in Ames, Iowa. Our meeting time is 2 PM on the third Sunday of each month of the year. Our membership dues are \$20 per year per family. Anyone interested in the Amiga is welcome to attend our meetings and is entitled to two free newsletters.

We are always very interested in trading newsletters and other pertinent information with other interested user groups. Interested parties should contact us via our club P.O. Box, or call me personally at (515) 233-5652.

Secondly, we'd like to compliment AC on a fine job covering the Amiga in the past years, and we're looking forward to many years of quality coverage in the future.

In a somewhat disgruntled tone, however, I would like to send a different message to the folks at Commodore-Amiga. Like so many big companies these days, they seem to be more interested in making money, not satisfied customers. While this may seem fine in the short run, in the long run it is the satisfied (or dissatisfied) customers (many of whom belong to user groups) that will continue to make a company grow



# IN SIDE-BY-SIDE COMPARISONS,



## LATTICE BLEW MANX AWAY.

### Introducing new Lattice C for Amiga DOS, Version 5.0.

*Lattice C 5.0 is the fastest compiler by every measure.* No other language, no other compiler, gives you faster, more efficient programs. And it now supports 68020 and 68881.

BYTE* Benchmark	Lattice C 5.0	Manx C 3.6	% Difference
dhampstones	42.4	62.7	32%
dhrystones/sec.	1605	1017	58%
matrix manipulation	15.7	50.4	68%
schigrd	70.2	82.8	15%
sorto	76.4	110.9	31%
sieve	34.4	40.5	15%

And no other compiler gives you a more complete programming environment. This package includes:

- Lattice C Compiler • Compiler Companion Utilities
- Source Level Debugger (CodeProbe) • Code Profiler
- Global Optimizer • 304 C Library Functions • Blink Overlay Linker • Assembler • Disassembler • Librarian
- Sample Files & Examples • Lattice Screen Editor • Installation Program • Comprehensive Documentation.

Our new, easy-to-use, Source Level Debugger, *CodeProbe*, supports both C and Assembly language, and multi-tasking debugging. And the new Global Optimizer will actually *enhance your programs' performance up to 40%*.

So the choice is easy: You can own the fastest, most powerful, most complete C programming environment available. Or for about the same price, you can buy a package that's shot full of holes.

*To purchase Lattice C 5.0 or for more information, see your favorite software supplier.*



## Lattice

Subsidiary of SAS Institute Inc.

Lattice, Incorporated  
2500 S. Highland Avenue  
Lombard, IL 60148  
Phone: 800/533-3577  
In Illinois: 312/916-1600

\*BYTE is a registered trademark of McGraw-Hill



# Amiga Genealogy?

## Now Shipping Your Family Tree

THE FIRST GENEALOGY DATABASE  
FOR THE AMIGA

- Standard Reports • Free Form Memo Area
- Supports Over 10 Million Family Members
- Easy To Learn and Use



## YOUR FAMILY TREE

Contact Your Local AMIGA Dealer or Send \$49.95  
+ 6% Sales Tax ( PA. Residents Only). \$4.00  
Shipping and Handling ( Within U.S.A.) to:

**MICROMASTER, INC.**

1289 Broadhead Road, Monaca, PA 15061

PH. (412) 775-3000

AMIGA is a Reg. Trademark of COMMODORE AMIGA

and prosper or fail and dry up. Therefore, it seems to follow that C-A should do all they can to provide information and support to user groups. After being an officer in a registered user group for over two years, I really question C-A's user group commitment. What do we get for the money we pay C-A to be registered? A few technical manuals that are of use to only a very small group of computer hackers! Oh, and if we are lucky, a bulletin from C-A once or twice a year. I think C-A owes it to the user group community to send out bulletins AT LEAST quarterly and to keep current user group lists available to anyone requesting them. I think C-A (and all the rest of the hardware and software developers as well) should also have reps visit user group meetings to let us know firsthand about the latest developments.

Thank you, AC, for letting me speak my peace, and if anyone asks me which Amiga magazine I would recommend, you know what I'll tell them.

Sincerely,  
Michael Drues

### A Colorful Amiga Fix

Dear AC,

Here's one for the kids as well as the adults. With the growing amount of software, this makes it easier to keep track of what runs on which Kickstart version and if it needs Workbench.

Each set of working copies of Kickstart, Workbench, and AmigaExtras have the same label color. After learning which group to use for each program, I draw a small colored box in the bottom right corner of the disk label to match its Kickstart's. Some get more than one box if any Kickstart works. I also write "Kickstart only" or "K.O." near the box for programs which boot themselves.

If the documents for your program do not specify which version to use, you can look for it in the startup-sequence.

Our 6-year-old is able to go from disk to disk without having to wait for Mom or Dad to tell him which Kickstart to use!

Keep those issues coming!  
Sharon Greig, Rosenberg, TX

*Continued from page 6*

Producing this magazine for three years has been more than fun. It has been an adventure I have shared with Amiga users around the world. I have been able to meet and talk with interesting people and see fantastic things develop.

It is also important not to become complacent with our popularity. Amazing Computing will continue to expand its coverage of the Amiga and maintain a balance of articles and features to best express the growth and popularity of the Amiga.

This issue marks not only the celebration of a three year anniversary, but more. It is the end of the beginning for both the Amiga and Amazing Computing. It is the start of a time to grow and produce more and better articles each month.

Our goal has always been to provide a wide diversity of Amiga information to the Amiga user. We have accomplished this through reviews, hardware projects, programming articles and more. Yet, we are continually searching for a better means to this end.

As always, the pages of Amazing Computing are filled with the efforts of fellow Amiga enthusiasts. We value your letters and submissions. Your contributions are the best way to add to the Amiga pool of knowledge.

We are searching for even more articles describing what you are doing with your Amiga. If you are accomplishing new and different things with your Amiga, or if you have utilized the Amiga's unique features in an exciting way, we want to hear from you. What may seem simple and everyday to you may be the inspiration a fellow Amiga user needs to complete their personal project. You are important!

Through the last three years, we have had a great time delivering the best of the Amiga to you. We have accomplished a great deal, but even more is yet to come.

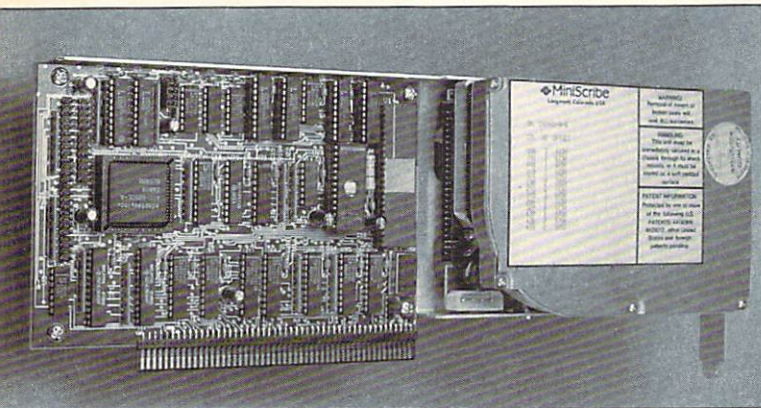
Sincerely,

Don Hicks  
Managing Editor



# HardFrame/2000

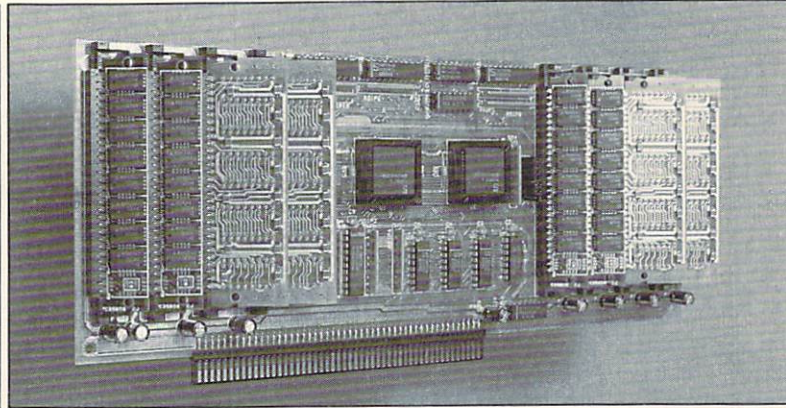
The Super-speed, DMA, SCSI Hard Disk Interface with 1.3 Autobooting



How fast is fast? **HardFrame/2000** transfers data at Amiga bus speeds! It's actually faster than the hard disk mechanism itself! And even more important in the Amiga's multitasking environment, **HardFrame/2000** has extremely efficient DMA circuitry to get on and off the bus in almost no time at all: 280ns to get on; 200ns to get off. **HardFrame/2000** autoboots under AmigaDOS™ 1.3 and is fully compatible with the new Fast File System. The core of any DMA SCSI interface is its SCSI protocol chip and DMA chip. MicroBotics has chosen the new, high performance Adaptec AIC-6250 SCSI chip, capable of up to 5 megabytes per second raw transfer speed, and the Signetics 68430 DMA chip running at 12.5 megahertz. Then we added additional FIFO buffering and enabled 16-bit wide data transfers for maximum throughput. The sophisticated design of **HardFrame/2000** provides for automatic SCSI arbitration, selection and reselection. The hardware supports either synchronous or asynchronous data transfer. **HardFrame/2000** can function as either the SCSI bus initiator or the target and can reside in a multiple master environment. Physically, **HardFrame/2000** is optimally flexible: the compact, half-size card comes attached to a full length, plated aluminum frame. The frame has mounting holes positioned to accept standard, 3.5" SCSI hard disk units such as those manufactured by MiniScribe, Seagate, Rodime, and others (hard disk mechanisms must be supplied by the user or his dealer as a separate purchase item). Alternatively, you can cable-connect to a SCSI drive mounted in your Amiga's disk bay or in an external chassis. As many as seven hard disks may be connected to a single **HardFrame**. There is no size limit on each disk. **HardFrame/2000** includes a 50-pin SCSI cable and header connectors for either 50-pin or 25-pin cable connection. Also included is a current tap to power frame-mounted drives directly from the slot itself. **HardFrame/2000** comes complete with driver, installation, and diagnostic software. Available September 1988. Suggested list price, \$329 (hard disk not included).

The **HardFrame/2000** photo shows the product with a MiniScribe 20 megabyte hard disk installed. Hard disks are not included in the purchase price of **HardFrame**. Note that if placed in the first slot, **HardFrame** uses only one slot.

# 8-UP! The Eight Megabyte Memory Card with Amiga-specific DRAM Controller Logic



All the memory space you and your Amiga 2000 need—in a modern, highly integrated FastRAM expansion board. In **8-UP!**, MicroBotics went all the way to provide you with a truly Amiga-specific memory design to meet the special demands of the Amiga's high speed multitasking environment: The heart of any memory expansion is its *DRAM controller circuitry*. Rather than compromising with off-the-shelf parts, MicroBotics developed its own, custom controller design and built it into high-speed, *Programmable Macro Logic* chips (Signetics PLH5501). These new, super chips (each **8-UP!** uses two PML's) permit MicroBotics to employ *sparse refresh* technology to assure that your **8-UP!** is a truly zero wait-state/minimal-refresh-collision memory design. If you're putting eight megabytes in only one slot, that means that you probably have plans for your other A2000 slots. **8-UP!** gives you new freedom to do that planning since, unlike other ram peripherals, it is an extremely low-power memory card—a single, fully-loaded, 8-megabyte **8-UP!** draws an astoundingly efficient 800 milliamps! That's less than *two-fifths* of the power "budget" for a single slot! Low power draw also means that the card is cool-running for reliability and long life (not to mention a cooler Amiga!). **8-UP!** offers you maximum flexibility in memory configuration: it is organized into two separate *PIC's* (Amiga-speak for autoconfiguring peripherals). Each **8-UP!** *PIC* consists of four SIMM module sockets; these sockets accept either 256k-byte or 1 megabyte *SIMM's* (Single Inline Memory Modules). You can also purchase optional *PopSIMM* boards from MicroBotics; fill them with conventional RAM; then use *PopSIMM's* to fill your **8-UP!** The card can run with as little as 512k of memory or as much as eight megs—with many intermediate configurations possible (particularly the six megabyte configuration, most desirable for use with a *BridgeCard™*). **8-UP!** is speedy, efficient, custom memory technology for your Amiga 2000—and it's available now! **8-UP!** suggested list price is \$199 (0k installed). Optional *PopSIMM's* are \$49.95 per pair.

The **8-UP!** photo shows the card half populated with conventional *SIMM* modules and half with MicroBotics *PopSIMM's*. *PopSIMM's* (without *DRAM* installed) are available as separate purchase items.



## MicroBotics, Inc.

Great Products Since the Amiga Was Born!

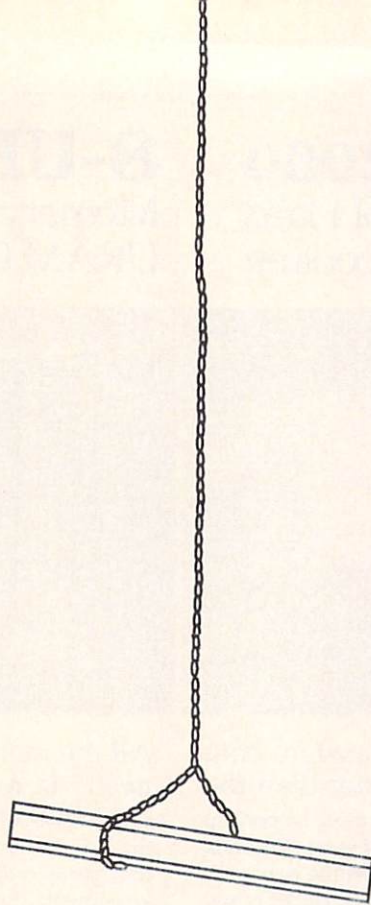
811 Alpha Drive, Suite 335, Richardson, Texas 75081 (214) 437-5330

SOLD ONLY THROUGH YOUR AMIGA DEALER

Tell your dealer he can quick-order from MicroBotics directly—no minimum quantity—show him this ad!

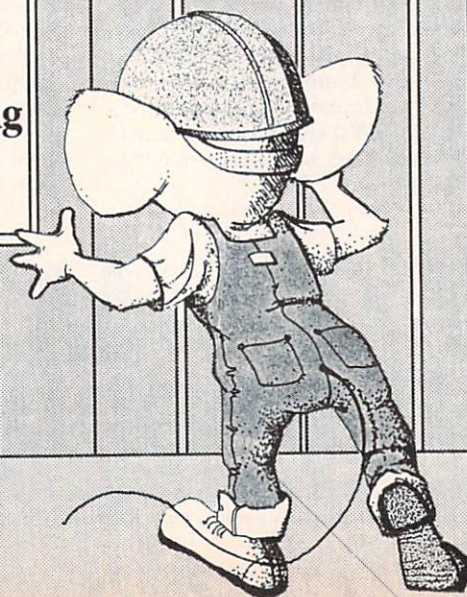
\*Amiga\* is a registered trademark of Commodore-Amiga. "HardFrame/2000", "8-UP!", "PopSimm", are trade names of MicroBotics





**Under Construction**

**Watch for something  
new and exciting  
on March 3rd**





# MovieSetter

### The Next Generation

by Steve Gillmor

Much has been made of the Amiga's graphics capabilities. For four years we've built on the innovation of Deluxe Paint, Deluxe Video, and the clever HAM wars of DigiPaint, Photon Paint, and PhotoLab. 3D, in its many incarnations, has pushed the limits of the hardware. The numerous animation products now available carve different slices out of the desktop video pie with a bewildering array of options and effects. As we watch Amiga animations produced with Zoetrope and Turbo Silver on the Jumbotron screen during SuperBowl Sunday, the proverbial Toaster looms enigmatically on the horizon.

I could go on and on, but the truth is if the Amiga is going to fulfill its promise, it must do with desktop video what Apple has done with desktop publishing: it must invent a world in which it can survive and prosper.

No program can be all things to all people. The limits of the current chip set dictate a series of complicated choices and trade-offs. MovieSetter is designed to run on all Amigas. This article begins with an Overview prefaced by some terminology.

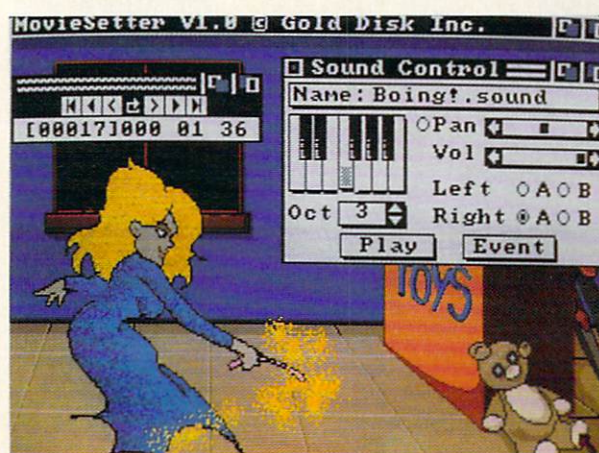
In brief, a Production is your MovieSetter movie (animation), composed of Sets of Faces (IFF bitmap brushes) animated over Backgrounds (IFF pictures), controlled on Tracks, and supplemented by Events like timing and palette changes, color cycling, scrolling and sound cues. You create these Productions in the SceneEditor, a kind of audio/visual word processor. You can jump over to the SetEditor to contour and order your brush groups in what amounts to a paint program for anims. You can import your character sets from your ClipArt disk and repaint and manipulate them in various ways, or create your own demos. Give them life in a series of motions, then send them back to the SceneEditor to be obliterated by your own customized RoboCop clone.

The Overview dissolves to the first of two tutorials: a studio tour of SceneEditor. First, load a Background, then choose a straight cut on from a requestor full of special wipe effects, and start your first track. Choosing Track/New tells MovieSetter you are about to add a new track and brings up the Set Load requestor. The program knows

what devices you have connected, from external floppy to hard drive partitions. Choose the familiar Boing Ball set from the data disk and the first Face appears attached to your pointer. You begin in the upper left corner and, in a series of clicks, Stamp the succeeding faces of the BoingBall set in an arc down to the road surface of the Background, then bouncing up and off stage right. Each click advances the set one face, adds a frame of animation to the movie, and advances you to the next frame for continued stamping. You complete your first Track by hitting F5 or by holding down the <Control> key when stamping the last frame. When you exit from creating a track, the program reactivates what resembles a VCR control panel which you can use to rewind to the first frame of the Production, and play by selecting the Play Forward gadget. The control window disappears and you see your ball bounce in a smooth arc from left to right.

After going to the frame where your ball "hits" the pavement, you choose "Sound" from the Event menu, bringing up the Sound requestor.

(continued)



(Left)  
Control your  
animation  
using a VCR-  
type panel

(Right)  
Insert stereo  
sampled  
sounds using  
the Sound  
Control  
requestor.



Double-clicking on the Boing! sound from the sounds directory places you in the Sound Control window. Soon you will be able to operate its variety of gadgets, but for now just click on the event button and close the window to register the event and return to play the production. You are immediately rewarded with the sampled sound of ball hitting ground in perfect sync.

At this point, you may be contacting your agent. You still have a bit more to do before lunch, though. The tutorial leads you through changing the "timing" in the event menu to slow down the ball from the default 10 frames per second to a stately 4 frames per second. You quickly add another ball and sound effect that follow close behind your first track. You are using the same set and sound, so you merely click OK in each requestor and reuse the already-in-RAM files, on their own new track and event.

Suddenly, the manual gets real, plunging you into the inner sanctum of track control. No need to be intimidated, though. You will find that this program is so well thought out and user-friendly that, in fact, the rest of the manual can be absorbed best by assembling a movie right away - no matter how simple it may be. You will find that the various commands and tools will reveal their uses more readily if you use them to navigate in and around your fledgling effort. I would also recommend loading the Demo Production from the program disk and examining it as you proceed with Track Editing.

There are two basic modes of track work: stamping down your sets, and editing the results. You will find many useful keyboard commands and shortcuts

to aid you in your placement and choice of set elements (faces) in track creation mode. Once you open a new track and begin clicking, you cannot move around from frame to frame in your movie until after you complete stamping. Once the set is attached to your mouse pointer, simply clicking to place the current face, advance to the next frame, and load the next face on your pointer. <LeftAmiga>-click is the same, but it switches to the previous face, and <Alt>-click is the same but does not switch faces.

*MovieSetter is so user-friendly that the manual can be absorbed best by assembling a movie right away— no matter how simple it may be.*

Here's a shortcut to save you from repeatedly clicking out a series of frames to "hold" a face in the same place. Select Track/Hold and enter the number of frames to Hold in the requestor. This command works like doing it by hand, but the program does the stamping for you and moves you to the correct frame to continue.

A similar time saver is Track/Repeat, which enables you to repeat a cyclical action (like a juggling sequence) with a global command rather than by continuing to stamp down faces over and over again. You can also automate face placement with a feature known as Guides. (I'll return to Guides after we're more conversant with the program.) You

can manipulate a few other editing features while creating a new track, but you can also access these while editing an existing track.

### Street Cleaning

By now you're probably interested in how to clean up that street scene you've no doubt gotten sick of. Of course, you could just go to the Production menu and Clear the whole mess, but let me recommend New from the same menu instead. Not only will it wipe the screen clean, but it also won't remove your background, sets and sounds from RAM. So you can now start again, only this time you'll learn how to edit an already created track.

After completing a new track, you return and gain access to the VCR play controller and the Track Edit window, which is a small grouping of tools used to reenter an existing track and make adjustments. These tools include:

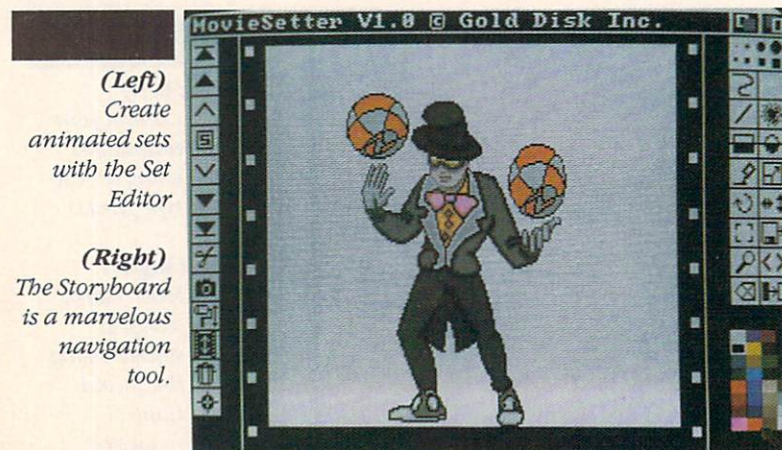
### Select Track

By clicking on its arrow button and then on a face, you select not just the face but the whole track. You can also access the Track/Select menu requestor and scroll to a hidden track (a track you can't see because it is behind another face) and select it from there.

### Move Track

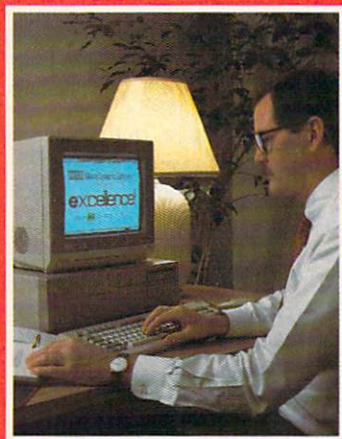
Selecting this hand button lets you reposition the visible track element on the screen. Holding down the <Alt> key while clicking will

(continued)





# Upgrade from Perfect to Excellence!



we compiled their suggestions and designed excellence!, a program that sets new standards for word processing. And more importantly, excellence! has been developed specifically for the Amiga, on the Amiga. It takes advantage of the user-friendly Amiga interface and is designed to be intuitive in a way no other word processor can match. An important point: several companies, new to the Amiga market, want you to think their track record with other computer systems makes them instant experts with your Amiga. That just isn't so. Micro-Systems Software is one of the pioneer developers of productivity software exclusively for the Amiga! We know your Amiga inside out. So, features you once thought to be luxuries, you can now consider basics. Excellence! has all the powerful features required of a modern word processor, in a package sophisticated enough to use in desktop publishing.

Micro-Systems Software is committed to a higher standard of excellence. And we're ready to prove it! Our newest Amiga product is a full-featured word processor that exemplifies our commitment to the Amiga. And to you. We have appropriately named it excellence! for obvious reasons. First of all, users of our popular word processor Scribble! told us about the features they wanted in a full-

featured word processor. So

There are always minor differences between programs designed for the same application. *Before* you make your choice, consider these major differences between excellence! and several well-known word processing programs! Excellence! processes words perfectly and does it faster than any other WYSIWYG word processing program available, giving the text-only programs a race for their money! (Not all programs claiming to be WYSIWYG really are. Excellence! shows you *everything*, including super- and sub-scripts, headers, footers, footnotes, colors, and graphics!)

Excellence! has all of the standard features too, including: mail merge, full clip-board support, full text styling, multiple methods of movement within documents, search

and replace, and printer control! Excellence! has a dramatic collection of features that place it at the forefront of a new generation of full-featured word processors. Its innovations include: full color support of text, inclusion of IFF graphic

Features	excellence!	Word Perfect v4.1	Visi-Write Desktop	Pro Write	Kind Words	Let Craft Plus
WYSIWYG (what you see is what you get)	✓	✓	✓	✓	✓	✓
Grammatical & Style Checker	✓	✓	✓	✓	✓	✓
PostScript Output	✓	✓	✓	✓	✓	✓
Include Pictures With Text	✓	✓	✓	✓	✓	✓
Fully Clipboard Compatible	✓	✓	✓	✓	✓	✓
Multiple Proportional Fonts	✓	✓	✓	✓	✓	✓
Color Support	✓	✓	✓	✓	✓	✓
Spelling Check As You Type	✓	✓	✓	✓	✓	✓
Math	✓	✓	✓	✓	✓	✓
Multiple Columns	✓	✓	✓	✓	✓	✓
Index Generator	✓	✓	✓	✓	✓	✓
Table of Contents Generator	✓	✓	✓	✓	✓	✓
Thesaurus	✓	✓	✓	✓	✓	✓
Limited Outliner	✓	✓	✓	✓	✓	✓
Mail Merge	✓	✓	✓	✓	✓	✓

images, spelling check as you type, basic math capabilities within documents, multiple column support, proportional font support, Index generator, Table of Contents generator, integrated Thesaurus, integrated Grammatical and Style checker, and PostScript output!

Excellence! is the perfect word processor for every need. From letters to newsletters; from the annual report to scientific research articles; from book reports to the book itself. If your writing aspires to excellence, you need a program to match!



Micro-Systems Software  
Committed to excellence.

12798 West Forest Hill Boulevard  
West Palm Beach, Florida 33414

Inquiries dial  
1-(800) 327-8724  
in Florida: (305) 790-0770



Amiga is a registered trademark of Commodore Amiga, Inc.  
PostScript is a registered trademark of Adobe Systems, Inc.  
Excellence! is a registered trademark of Micro-Systems Software, Inc.  
Committed to excellence since 1978.



globally reposition the entire track's faces by the same relative amount. If you want to reposition a set to start offscreen in overscan when played, you can hold down the arrows and move the selected face (or globally the whole set) in any direction out of frame.

### ***Insert Elements Before and After***

Automatically puts you in track creation mode either before or after the current frame.

### ***Track Behind and In Front***

When you create a track, it is initially placed in front of every visible track. These gadgets, similar to Workbench depth gadgets, allow you to move tracks either in front of or behind other tracks. All tracks remain in front of your background.

### ***Copy, Cut, Paste Track***

These powerful commands allow you to copy, remove, and paste existing tracks into your production in other spots, or at the same time create multiple clones of the same effect.

### ***Change Face***

Allows you to cycle through the faces of the currently selected track in either direction.

### ***Go To Beginning and End of Track***

Self-explanatory.

There are keyboard shortcuts to delete the current face, or all previous faces, or all future elements. Finally, there is Shift which is accessed from the Frame menu, where I will discuss it and other global commands that perform similar actions.

Recall that I suggested examining the Demo movie on the disk as you learn your way around the Studio. There is a marvelous navigating tool in MovieSetter, the Storyboard. This Production menu option gives you a visual database of all the individual pieces of your movie. It's a sort of MicroFiche Filer Jr. with the "key" frames of your animation displayed in small form in a series of nine panels on a resizable window with its own Conditions menu. The menu sorts the production according to the events that you choose. If you enable Sound, the Storyboard window creates a frame corresponding to every sound event, and when you click on the frame you move immediately to that location in your movie.

The Storyboard defaults to Track Start and Background Change. Other choices on the Conditions menu include:

- Scrolling
- Color Cycling
- Palette Change
- Timing Change
- Loop

Only nine frames are shown at a time, but you can hit <UpArrow> to show the next nine frames, and so on. You can leave the window on the screen

and return to shuttling through your movie by clicking anywhere other than in the Storyboard window. This switches you back to the full menu so you can access other Production, Frame and Event menu choices.

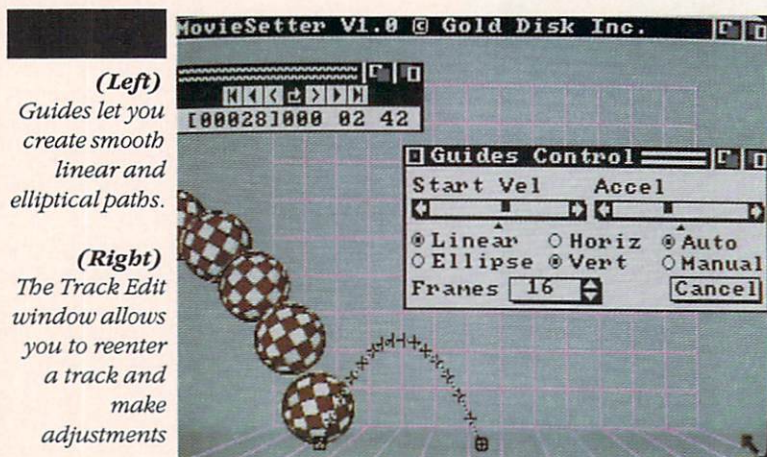
### ***The Event Menu***

While we're on the subject of events, why not run them down quickly? MovieSetter lets you add sampled stereo sounds. In the Sound Control window, you can change the pitch, octave, stereo positioning (of mono voices) and volume by clicking on various gadgets. Naturally, you can only play four mono or two stereo sounds at one time, but you can minimize the memory used by a single sample by reusing it at various pitches and volumes to simulate different sounds. For example, the sound of a church bell raised several octaves and three half-steps on the piano key gadget suggests an elevator bell (as in "Ding").

When synchronized properly with animation, sounds can convey the illusion of motion. Experiment with moving a sound effect around and you'll see that subtle shifts in volume, timing, and pitch can work wonders.

Color Cycling can be turned on and off at various frames, and run at different speeds with up to four cycles running simultaneously. You have to be careful not to cycle a color that is in faces or parts of the background you want stable. Cycling is not easy to master, but if you reserve enough colors for the effect, it can save you in memory consumption as a pseudo-animation effect.

(continued)



**(Left)**  
Guides let you create smooth linear and elliptical paths.

**(Right)**  
The Track Edit window allows you to reenter a track and make adjustments





# Unchain Your Amiga



## Get Amiga 2000 performance without Amiga 2000 prices.

Now you can add Commodore's A2090 hard disk interface, the A2088 IBM BridgeCard\*, Expansion Technologies Flash!Card HD interface, memory expansion, or any other A2000 specific cards with your Amiga 500 or 1000.

By providing two vertical 100 pin slots the ToolBox lets you use Amiga 2000 cards with your A500 or A1000 computer. It also provides you with a power status indicator light, an all metal chassis, and a bus pass-thru so you won't be chained down in the future. With its flexible design you can choose a model with or without a power supply to match both your needs and your budget. What's more it has Expansion Technologies one year warranty on parts and labor.

### Models available:

Model #	Comp.	Power	Dimensions	Power Options	Price
EEC-2510	A500	1 amp @ +5V	4 <sup>3</sup> / <sub>8</sub> x 6 x 14 <sup>3</sup> / <sub>4</sub>	The Brick*	\$189.95
EEC-2520	A500	2 amp @ +5V 1 amp @ +12V	7 <sup>1</sup> / <sub>4</sub> x 6 x 14 <sup>3</sup> / <sub>4</sub>	Not required	\$259.95
EEC-2110	A1000	None	4 <sup>3</sup> / <sub>8</sub> x 6 x 14 <sup>3</sup> / <sub>4</sub>	1 amp off bus *	\$179.95
EEC-2120	A1000	2 amp @ +5V 1 amp @ +12V	7 <sup>1</sup> / <sub>4</sub> x 6 x 14 <sup>3</sup> / <sub>4</sub>	Not required	\$259.95



Now with all these models it's easy to link your A500 or A1000 to the wide variety of A2000 cards available, but if you're having trouble choosing give us

a call. We'll be happy to give you more information on the ToolBox or any one of the other great products that Expansion Technologies offers for the Amiga family of computers.



**EXPANSION**  
TECHNOLOGIES

46127 Landing Pkwy.  
Fremont, CA 94538  
415/656-2890

\*Call for complete information (no chain mail please).

Amiga 500, 1000, and 2000 are trademarks of Commodore Business Machines Inc.



Palette changes can be event-driven as well. You can access a typical palette requestor, complete with spread, copy, exchange, undo, and a choice of both RGB and HSV sliders. Turn any corner in this program and you find yourself in another familiar Amiga environment. You can use palette changes to easily turn the sky in the tutorial background from day to night, then drop a car set tooling down the highway with scrolling enabled. A neat effect, done in ten seconds by lowering the luminance control for the blue color of the sky. We'll explain Scrolling when we talk about Guides, as they both employ similar requestors.

Rounding out the Event Menu are Timing and Loops. These events can be changed as often as you like. If you select the last item on the menu, Select shows all the events in the current frame. If you step through the movie, the list of events will update automatically. This is a great way to decipher MovieSetter animations you download from PLink and other online services.

Similarly, you can use Track/Select (called Edit in the manual) to list all tracks visible in the current frame. This aids you in selecting an obscured track for editing. Other items on the Track Menu include the previously discussed New, Hold, and Repeat. You can Delete the currently selected track, as well as rename sets with Name.

### **The Frame Menu**

The Frame Menu gives you commands to Duplicate, Delete, Shift, and Add Start and End frames to your movie. Duplicate is a way of basically creating a freeze frame of a scene, where you just repeat the whole shot for some time, such as just before "The End". Add Start and End are useful for preparing the beginning or end of your production for new backgrounds or clean starts and ends. Shift is a toggle that works in conjunction with tracks. When enabled, any track that is added to a movie will cause all future tracks and events to shift and occur later. (The word processor analogy is particularly appropriate here.)

This is similar to Insert on the Production menu, used to insert scenes previously saved inside an existing production. In fact, with an Insert, the program performs a shift transparent to the user, so all the rules for Shift apply here as well. Insert also has the advantage of not reloading duplicate sets, backgrounds, and sounds. You can use this to save different versions of productions using the same parts; when you Insert on the first frame it is identical to Loading a production from disk, but since the various parts are not Cleared from RAM, the program quickly inserts the MovieSetter script that controls the various files.

Other Production Menu choices include Save Embed, Save No Embed, and Components. When MovieSetter creates a production, it writes its own

script file containing all the event information. This "blueprint" is usually a small file, so Save No Embed is a quick way of backing up your productions as you create and edit them. Save Embed saves the script along with all the data files. Components is a way of disassembling embedded productions and automatically saving the separate sets, backgrounds, and sounds.

### **The Special Menu**

Speaking of saving, I've saved the Special Menu for last. It lets you toggle History, Borders, Wipes, Cycling, and Interlace off to aid in your editing, and back on for post production. There is also a Select option with subitems for Sets, Backgrounds, and Sounds. This is used to strip unwanted files from RAM, a helpful housekeeping option

That leaves Guides and Set Editor, which is MovieSetter's secret passageway to its combination prop, wardrobe and makeup departments. This is like another program inside MovieSetter. In fact, the two Editors can be run separately from the Workbench, which is the best option for 512K users. Set Editor is like DPaint, Jr. in that it has most of the tools of this granddaddy of Amiga graphics ranging down the right side of the screen. It does have a few extra touches of its own worth mentioning, including a resize tool that is a very flexible workaround to the lack of any auto-sizing "tweening" effect in the Scene Editor. The Magnify tool has a unique way of scrolling that is also

## **MovieSetter & Comedy**

The release of MovieSetter has come at a good time for us. My partner, Tina Chase, and I have been commissioned to do a promotional video for Catch A Rising Star, the nationwide comedy and music club chain. The video will be used to promote the chain's club at the Chicago Hyatt Regency Hotel, and will be played on the in-house channel of the hotel's in-room cable system. It will combine Amiga graphics, sound effects, and music with live footage of your favorite wise guys and gals.

MovieSetter offers a perfect platform for this project. Its ability to store and play MINUTES of animation

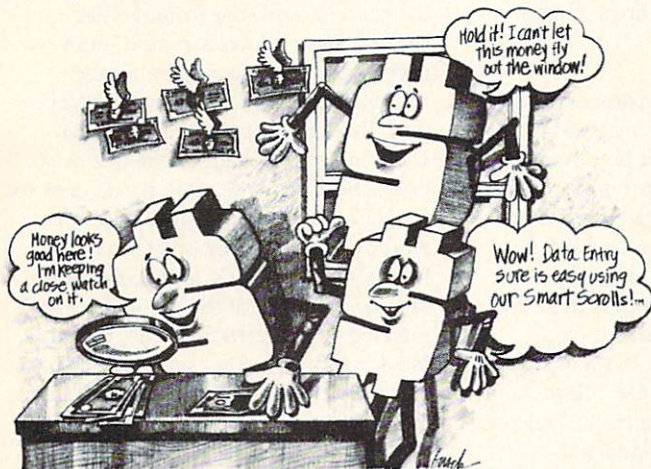
and audio in overscan helped seal the deal when we made our first client presentation. We're employing the full array of Amiga graphics products, using Interchange to move between Sculpt 4D, Turbo Silver, Modeler 3D and others. A new Hash Enterprises product, Animation:Editor, is used to convert from various anim formats and screen sizes to MovieSetter's resolution. Fonts from Masterpiece Professional Font Collection are used to create a takeoff on the old Twilight Zone title sequence, loaded into Zoetrope, shattered and spun. Zoetrope, however, does not currently support

overscan, so we capture the anim as a series of brushes. Using Load IFF from the SetEditor Element window, we load the brushes into the SetEditor, with registration marks already in place using Insert Face lower mode. Then it's a simple task to stamp away and watch the letters form into the Catch A Rising Star Zone-ified logo with some familiar snatches of music created with Soundscape multi-tasking with DMCS, sampled with AudioMaster II and loaded into MovieSetter. You get the idea; this marriage of computer and comedy is made in the shade.

—Steve Gillmor



## Meet a team of the friendliest financial organizers you'll ever run across.



**When you want to manage your personal finances, Money Mentor goes a step beyond.**

Plug Money Mentor into your Amiga and a virtual teamwork effort takes place in watching over every aspect of your personal finances.

The new "C" version of Money Mentor is the friendliest financial organizer obtainable today!

Now you can experience super-speed data entry, dazzling graphic output and an extremely friendly attitude!

### Smart Scrolls for speed.

Money Mentor has a truly unique system called *Smart Scrolls*, that handles a diversity of otherwise tedious data entry functions and clips along saving you up to 70% of your typing time. It's a *smart* addition to Money Mentor, that's why we call it *Smart Scrolls*.

### Money Mentor Features:

- Net Worth Statement
- 200 Budget Categories
- 30 Integrated Accounts such as Checking, Cash, Savings and Credit Cards
- Elaborate Search Routine allows editing of transactions according to your specific guidelines
- Automatic Check Printing
- Automatic Account Balancing
- Color Graphic Reports illustrating *actual vs. budgeted* amounts
- Over 50 Reports to choose from!

### What they're saying about us!

"Money Mentor has to be the nicest look and feel of any money manager package for home use that I have ever seen." — Amiga Sentry

"Money Mentor is an excellent product"

— Amazing Computing

### Money Mentor is for everyone!

It does more than just keep your checkbook balanced. Money Mentor helps you manage your personal finances which is important to any family or individual.

With Money Mentor, you can be looking better financially.

Order Money Mentor today.

**Money Mentor sells for only \$95.95!**



SEDONA SOFTWARE/11828 RANCHO BERNARDO RD., SUITE 128-20/SAN DIEGO, CA 92128/CALL (619) 451-0151

implemented in the Clipboard Window, where you can import IFF pictures and clip brushes back into the set editor.

The left side of the Set Editor screen holds tools to control the creation of the animated sets. The familiar VCR-like controls are there, as are the Copy, Cut and Paste, Insert face, and Registration Mark controls. Registration marks are used by MovieSetter as position guides to snap faces to when you stamp down your sets. You can switch these marks on and off with Show Registration Marks.

You'll notice that in the Set Creation tutorial, you start with a very simple stick figure, selecting the registration mark and centering it on the character's head. Selecting the Copy Element tool places a duplicate of face number 1 AND the registration mark in the buffer. Both Paste and Insert face are toggled, and by clicking on the lower

half of either, you carry over the registration mark in the same position throughout your faces as you create your new set. This is a handy way of automating the importation of a series of brushes created in other programs. (See SideBar)

Having created, tested and saved your stick character, you are ready now to jump back into the SceneEditor, selecting Set/Exit to return. The set you've been working on will be available to you for stamping back in the scene, and by now you're ready to tackle Guides. By the way, it's good practice to save your new or modified set before you leave the SetEditor; it's easy to forget, and if you fail to Save Embed your production and save just the script at the end of your session, you'll lose your Set work. If you did Save Embed, use Components to extract your Set, and Save As under a new name.

### Guides

Now for Guides. Although accessed via the Frame menu, guides are first mentioned as "the most powerful track creation tool." Guides let you automatically create smooth linear and elliptical paths, affected by both velocity and acceleration. First, you select Track/New and choose a set. When you have a face attached to the mouse pointer (at the Registration mark) select Special/Guides and the Guides Control window appears. At this point in the manual, it is easy to become a bit confused, and in a few paragraphs you might feel that it's easier to learn about this another time. You are advised that "The easiest way to understand the operation of these tools is to experiment." True, but the first step is the hardest. Here's a mini-tutorial that might begin to shed some light on this tool.

For starters, let's use the default settings of Linear, Vertical and Auto. Click in the Frames gadget, erase the 10 frames setting and make it 30 frames

(continued)



## Multi-Forth

### The Language of Innovation

If you haven't tried Multi-Forth™ you may not have yet unleashed the full power of your Amiga. This comprehensive development environment includes:

- Local Multi-Tasking
- Built-in Assembler
- Turnkey Compiler (royalty free)
- Sound Drivers
- AmigaDos & Intuition Support
- Complete Set of Include Files
- CompuServe Public Forum
- Extensive Documentation
- PowerWindows Option
- New price - \$99

Call for a technical data sheet or check out our online services on CompuServe at GO FORTH.

#### *Creative Solutions, Inc.*

4701 Randolph Rd. Ste. 12

Rockville, MD 20852

301-984-0262

1-800-FORTH-OK (367-8465)

Major credit cards accepted

instead. Hit <return> and move your mouse up to the Start Velocity gadget. Click on the right arrow of this gadget 3 times, then drag the whole window up by the drag bar to give you room to work with in the main window. Click in the main window and draw a one-inch horizontal line. Release and you should have a two-inch line of dots, each dot representing a position where a track element will be automatically stamped. If you aren't happy with it, just click again in the window and draw another line.

Now click on the same right arrow of Start Velocity two more times. The line should lengthen towards the right. Now move over to the right to the Acceleration gadget and click ONCE on its right arrow. The line should turn up at its right end. Click on the Accel left arrow twice, once back to center, then once to the left. The line should go down. Now click on the little triangle in the center just below the Accel gadget (just above the word Auto) to center Acceleration at zero. Change from Vert to Horiz by clicking in the little circle to Horiz's left, then go back to the Accel

gadget and repeat the actions, first clicking once on the right arrow (the line should lengthen to the right), then twice back the other way (the line should shorten to the left). What you are seeing could be likened to the effects of gravity, a useful analogy since animation works best by exaggerating the laws of nature.

#### **Ellipses**

OK, what about ellipses? The main function here is to create a curved arc. Here's a way to create the time-honored circling stars effect of a hammer to the head. Leaving Horiz activated and Start Vel where it is—five clicks to the right (positive)—center Acceleration again. Then click on the left arrow of the Accel gadget TEN clicks, and change from Linear to Ellipse. Clicking in the main window, draw out a circular ellipse about two inches in diameter and release the button. Now set the band still attached to your pointer to 3 o'clock to select your starting point on the ellipse, and click. You should see a curved line that travels about a fourth of the way around the circle from about 3 to 6 o'clock. Now move over to the Start Velocity window and start clicking on the right arrow. The curved line should begin circling until you've come around full circle. Now all you have to do is press the Close Window button to exit Guide Control. "Auto" tells MovieSetter to create the specified number of frames along the guide, automatically cycling through your track's set of stars or birds, whatever. If you Copy Track, you can then Paste it down over and over again to create that effect of circling objects.

That should get you started with Guides. As you learn to manipulate these paths, you will learn how to create slow-ins and outs and other tricks of the trade. Try figuring out the physics of a double take and assembling an ellipse whose parts you could use for that anticipation, action, and reaction effect. Go to Manual, and you can use the guide as a template for snapping to, and hit <g> to toggle the Guide on and off.

Accessed from the Event menu, Scrolling is another tool to master through practice. The top half of the Scrolling window has two gadgets: Start and End Velocity. The lower half deals with Acceleration. The Start Velocity represents how fast we go initially, the

End the top speed we will reach. Accel represents how long it will take to go from start to end velocity.

#### **Test Drive**

Here's how to create the effect of a car speeding up and slowing down on the road background from the first tutorial. Load the background, then access the Scrolling subitem of the background menu item. Set Start Vel to Zero by clicking on its triangle, then click on End Vel's left arrow five or six times. Now set Acceleration one click to the left, click on Event and close the window. Select a New track, the car.set from the ClipArt disk, and stamp down fifty or so frames of the car in the SAME position. Now go to frame 25, reenable the Scrolling window, set both End and Start Velocities to Zero, then click Accel one to the right. Leaving Start Vel at Zero, click on Event, close the window, rewind your movie and view it. You should see the background speed up then slow down. You might see a jump in the background at some point in the middle, but I'll leave that one for you to figure out.

•AC•

#### **So you wanna be in pictures?**

So do a lot of people, including the Fortune 500. The Amiga is garnering some good press these days in the legitimate video business, particularly in the area of character generator software. In business, corporate annual reports are being produced on video; The dollars spent on corporate video have gone from \$800,000 in 1983 to a projected \$1.2 billion in 1990. When Wall Street discovers the speed, facility, and cost effectiveness of state-of-the-art products such as MovieSetter, they will join us as we Boldly Go Where No One Has Gone Before.

#### **MovieSetter \$99.95**

##### *Gold Disk*

P.O. Box 789, Streetsville  
Mississauga, Ontario  
Canada L5M 2C2  
(800) 387-8192 US  
416-828-0193 Canada



# SuperBase PROFESSIONAL

*A User's Perspective*

by Marion Deland

Superbase Professional is a powerful professional database with a separate forms editor, a programming language, a built-in text editor, "VCR-style" controls and, in the newest upgrade, a telecommunications option.

Precision Software Inc., who now market the program themselves in the U.S., present Superbase Professional as an applications environment, and that's how I got interested in it. I needed a relational database for a particular business purpose—analyzing publicity placements for a client. It was a project that I hoped would grow into a service business, and I needed a program that would adapt and grow with me. Superbase, first Personal and then Professional, seemed to be the answer.

Before we go any further, let's get clear which Superbase does what. The system is modular; as you go up the line, more modules are added. This is how it lays out.

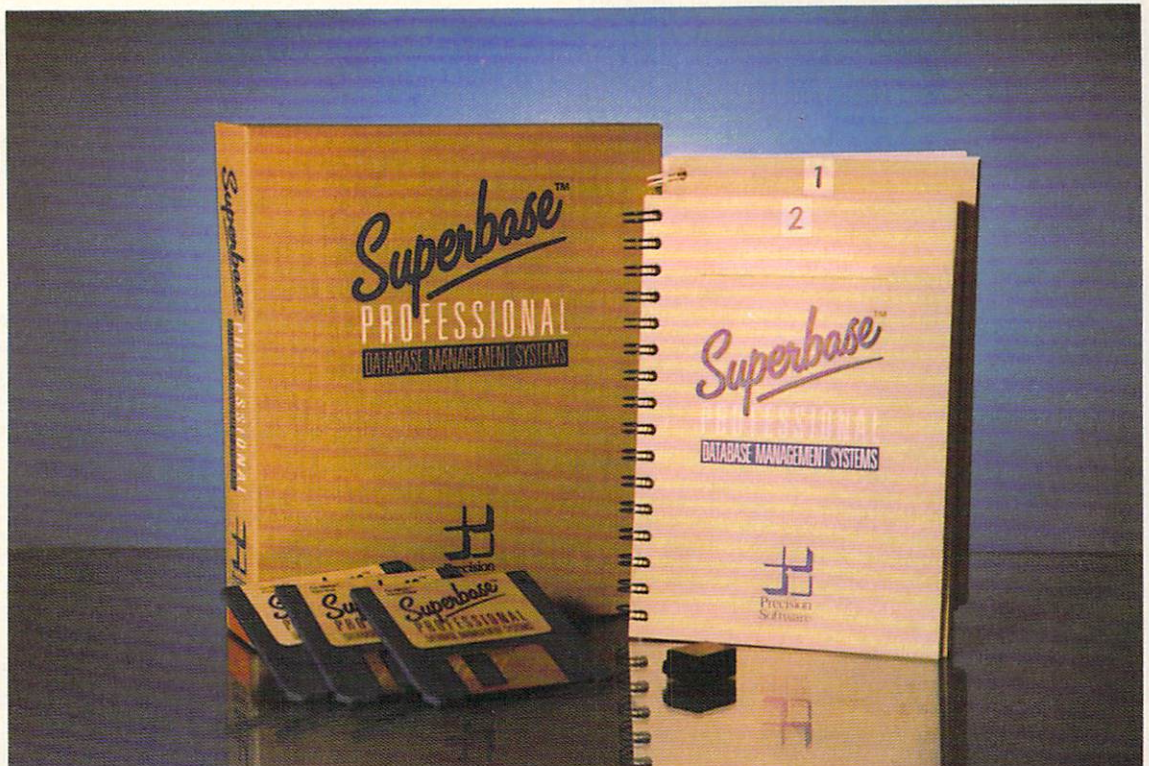
**Superbase Personal** is the "core" module, with data entry, three view modes: record, table and form (not to be confused with forms created with the Forms Editor in Superbase Professional), and a query function to report your data. **Superbase Personal 2** consists of the core module plus text editor and telecommunications.

**Superbase Professional 3** includes the core module, text editor, telecommunications, plus the Forms Editor and DML (Database Management Language).

According to Precision, these three Amiga products will all remain on the market. Registered owners can "trade up" for the difference in price, keeping their old program, which they can then resell. Each program will be upgraded as changes are made to the modules they contain, and upgrades are available to registered owners for \$10.

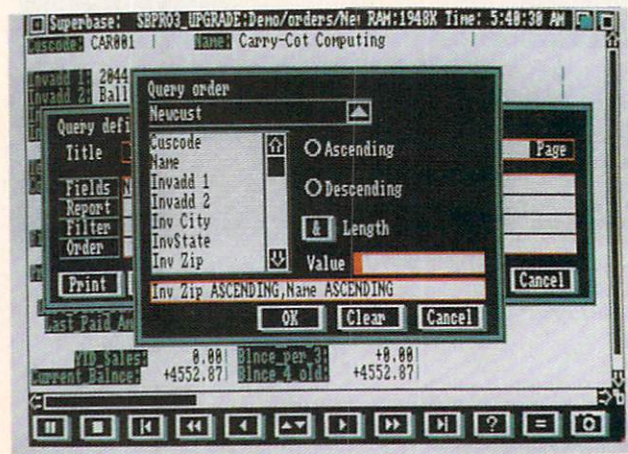
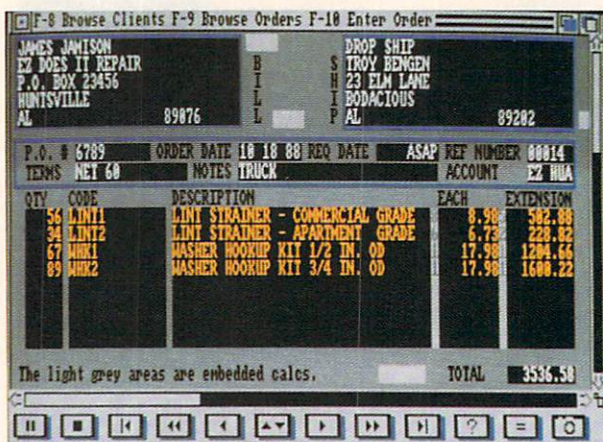
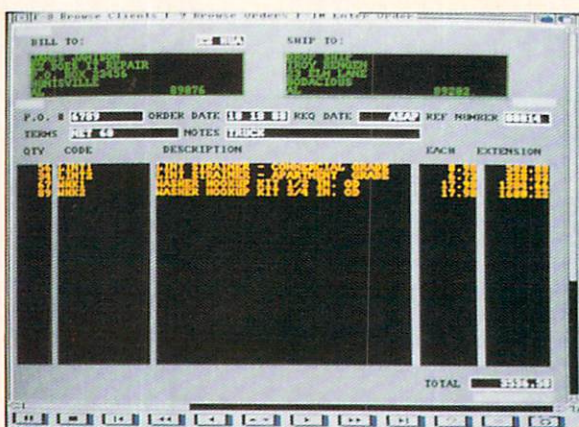
### Superbase 4

You may also see a Superbase 4 around. It is a version of Superbase Professional 3 for the PC, running under the GEM operating system. It is file- and program-compatible with SBPro 3.



(continued)





## Expectations

We've come to expect a lot of Amiga programs. We take things for granted, like the intuitive use of mouse and menus. With our disks full of IFF files, we expect a database to be able to handle graphics and sound.

We also want it to be as powerful as any business database management system in the PC marketplace. I had some specific requirements for my own purposes. The program had to be easy to learn. It had to be fast. I needed speed in accessing, saving and finding records, and in reporting.

My database needs were still growing, so I needed flexibility, in file size, and in the number and length of fields. And I needed to be able to change the file definition and add fields at a later date, after records had been entered.

I also needed a variety of field types, with validation options. And because I might have others helping me, I needed—with my elementary programming skills—to be able to control and simplify data entry as much as possible.

I needed access to more than one file at once, and to files in more than one directory. And I wanted to install the program on my hard disk.

I wanted to be able to get "quick and dirty" reports for my own use, as well as a final, formatted one for the client. I would need to be able to import and export data from and to other programs, both mine and my client's. Since I might need to send data by modem, I wanted telecommunications. Finally, I had to be able to protect my database from careless (or deliberate) deletions and changes.

So how well did Superbase Professional live up to all these expectations? Remarkably well.

## Learning a step at a time

This heavy-duty program succeeds in being relatively easy to learn, by "layering" its complexity. You can start out the easy way, building files with menus and mouse clicks and reporting the results with the query requester. As you get more comfortable with the program, you experiment with the forms editor and the programming language. Stay with it, and you find yourself able to write complete applications.

Help is provided via demo files and two easy-to-read manuals (plus supplements). Also, Precision Software, now marketing the program themselves in the U.S., is putting a lot of emphasis on technical service and support. This is needed. When Progressive Peripherals marketed the program, they couldn't live up to their promises in this area.

The Official Superbase Information Network (OSIN) is now a section of American People/Link; and there will also be a knowledgeable voice on the other end of the phone at Precision. An example of the technical information now being provided by Precision Software is a detailed listing of file formats in the new "Superbase Journal", available by subscription to registered owners.

If you're interested in developing applications, Precision wants to hear from you. They will help in developing and marketing applications. Call during business hours or contact USER ID: PRECISION on the OSIN network. Ask about OSAD (Official Superbase Applications Developer program).

Now, about those manuals. First, let me congratulate Precision on the writing—it is clear, logical, informative—even funny! However, the organization can be confusing. Because the Superbase line is also marketed for Atari and IBM (under the GEM operating system) computers, the manuals are "bi-lingual". (Somebody told me he'd gone through his entire 2 volume set, yellow-marking the Amiga references!)

(continued)



**Color Graphics.**

**Multiple Fonts.**

**Spelling Checker.**

**Mailmerge.**

**Multiple Windows.**

**Resizable Graphics.**

**Fast Printing.**

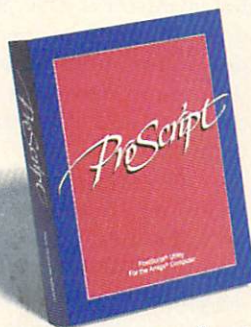
**WYSIWYG.**

**No Other Word Processor For Amiga®  
Stacks Up To ProWrite™ 2.0.**

Word is out. ProWrite 2.0 allows you to expand your Amiga word processing to new horizons. Use it to generate proposals with impressive bar charts. Create letters and reports with snappy graphics. Generate television storyboards. Even dash out newsletters with spot illustrations.

And of course, ProWrite 2.0 has all the other features you expect: a spelling checker with 95,000 word dictionary. Mailmerge. And fast graphics printing. Best of all, it works with our font packages, ProFonts I and II, giving you a broad range of professional and decorative fonts.

All this, plus ProWrite's powerful editing capabilities and ease-of-use combine to make ProWrite the best word processor for the Amiga computer.



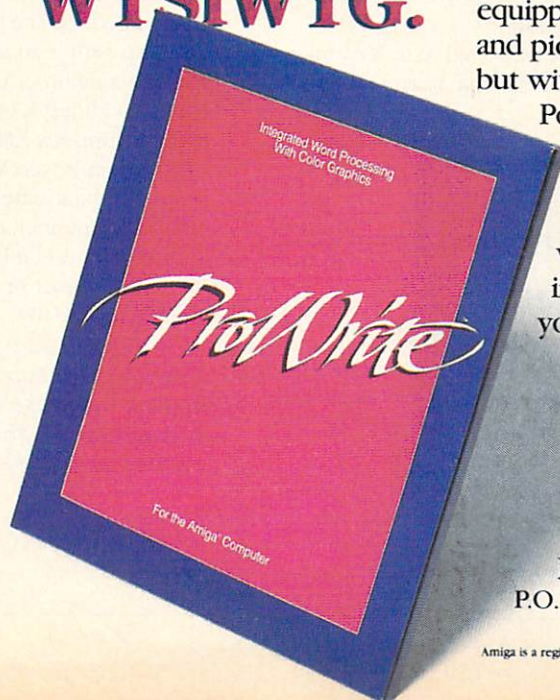
**ProScript: Perfection In  
Print For The Amiga.**

ProScript is the new PostScript utility from New Horizons Software that gives you the professional look of typeset copy.

ProScript reads ProWrite files

and prints them on any PostScript equipped printer, giving you the same multiple font and picture capabilities you expect from ProWrite, but with the typeset quality of PostScript printing.

ProWrite 2.0 and ProScript. A powerful new team that puts a new world of word processing and printing at your fingertips.



**New Horizons**  
S O F T W A R E

**First in Personal Productivity and Creativity**  
P.O. Box 43167/Austin, Texas 78745/(512) 328-6650

ProWrite, ProFonts and ProScript are trademarks of New Horizons Software, Inc.  
Amiga is a registered trademark of Commodore-Amiga, Inc. PostScript is a registered trademark of Adobe Systems, Inc.



Also, since the programs are modular, adding features as you move up the line, the manuals consist of several sections, each indexed separately. A single complete index would be a valuable aid for users of SBPro.

I had another problem with the indexing, one that is by no means unique to Superbase. An index should be more than a rearranged table of contents, it should try to predict the reader's approach and direct them to the answers. Here's an example of the problem.

I wanted to get rid of a record. I looked under "Delete", then under "Erase", then finally under "Record", where I found the term was "Record remove". To get rid of a file, however, I had to look under "Deleting files". Putting more thought into the indexing would do wonders for the learning curve on Superbase Professional.

### Speed-up options

The next factor that interested me was speed. As my files grew, would SBPro be able to keep up? So far, the answer has been yes—and one of my data files has passed 880K. Storing files in RAM, I have developed multi-file queries with multiple sorts that popped up results in just a few seconds.

Superbase Personal takes the safest approach to data entry, opening and closing files with every new record. If you're working with floppies, this can get a little tedious, and SBPro has added a "batch entry" option to speed things up. It also copies files into RAM and back again for you, though in early versions of SBPro I had problems—the occasional crash—using this method for data entry. I now play safe and use it just for reporting, and for that, it's excellent.

### Flexibility

SBPro is flexible. You have lots of choices, and you can change your mind later. If you load SBPro from the CLI, for example, you can give it parameters that eliminate the control panel, sizing gadget and/or scroll bars, give the program its own custom screen, disable chr\$(13) or load a form at startup. SBPro 3, in fact, can be made invisible to the user.

When you first create a file in SBPro, you select a data type for each field, choosing from Text, Numeric, External, Date, and Time. You can define calculation formulas, constant formulas and validation formulas. A SER function reports the number of records in a file, and a ternary operator even lets you build "IF THEN ELSE statements" into calculation formulas.

External graphic fields are displayed on a separate screen unless they appear in a form, where they can be any size or position.

One of the things I like most about Superbase is that you can edit the file at any time, even after you've begun entering records. If you delete a field, it stays in the file definition until you REORGANIZE the file. Strangely, there doesn't seem to be any way to restore a deleted field from within Superbase, but you are protected from accidentally deleting a field after you've entered data in the file.

File size and number of fields are limited only by memory. You can have up to 999 indexes, either unique or "normal". (When you first create a file Superbase has you create an index before you save the file definition.) Index files are stored with the data and the file definition.

You are restricted to a maximum field length of 255 characters, though

one field type allows for an "external" text file (or IFF graphic/IFF sound file/sound data dump) of any length. This might present difficulties in importing a file from a program like dBaseIII, which allows for larger memo fields.

Speaking of importing/exporting files, SBPro 3 will convert files to and from dBase, Lotus, Logistiks (also a Precision program) and .DIF files, as well as ASCII. Superbase 3.0 also provides for compatibility with ARExx, the macro language that promises to add a new dimension to multi-tasking on the Amiga.

### Multi-file operations

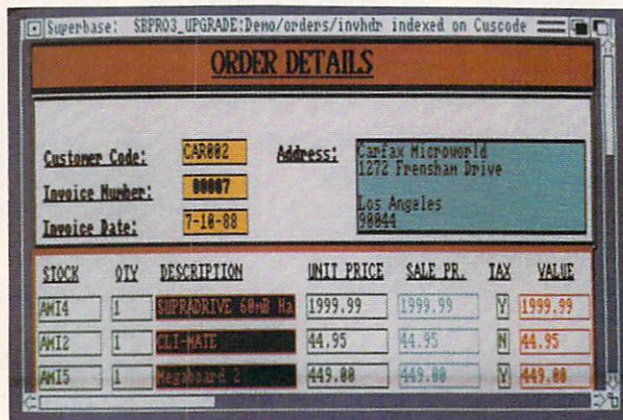
By definition, a relational database must be able to open several files at one time. With Superbase Professional, the number of open files is limited only by memory and common sense. (The fewer open files SBPro has to deal with, the quicker it will generate reports.) Multi-file queries, forms and updates can be loaded from disk, and they will in turn open the files they need—a real time-saver. You can also create a "start" file to set up your application before you start.

Having at last invested in a hard drive, I wanted to be sure that Superbase Professional could be installed. I did it the easy way, with the Workbench icon, with no problem. The program is dongle-protected, but registered owners can now get a backup copy encrypted with their name, address and registration number—and no dongle protection.

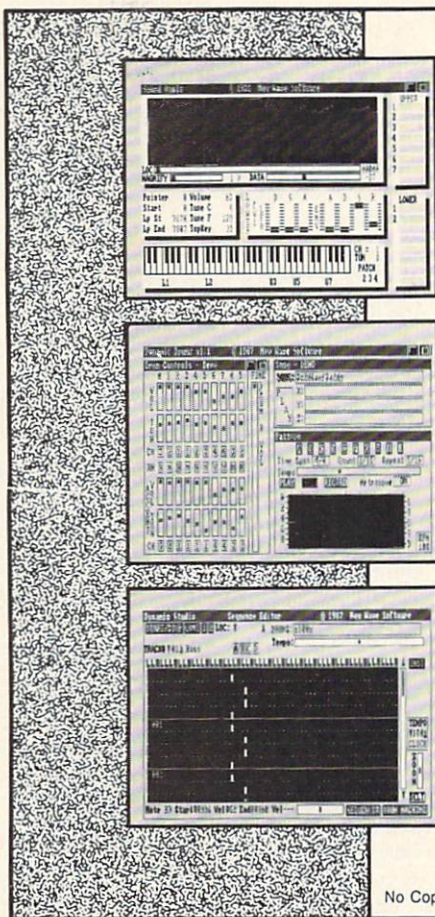
### Telecommunications

Version 3.0 includes telecommunications, as well as several enhancements. The telecommunications option is simple to use, operating either through a menu or under program control with just a couple of lines of code. It supports XMODEM, XMODEM-CRC and improved WXMDEM protocols, baud rates of 300, 1200, 2400 and 9600, and includes autodial and autoreceive. A File Header option allows either the sender or the receiver to select a name for the file.

You can send a file to another Amiga running Superbase or to a PC running Superbase 4, and automate it to be handled by Superbase in your absence (at cheaper phone rates).







## Sound oasis

\$99<sup>95</sup>

## dynamic drums

\$79<sup>95</sup>

## DYNAMIC STUDIO

\$199<sup>95</sup>

The internal sound capabilities of the Amiga are better than that of any other personal computer. These capabilities mean nothing though, without quality digital sounds, which up till now have been scarce. Sound Oasis gives Amiga owners access to a large library of studio-tested digital samples, by using the Amiga's built in disk drive to read disks made for the Mirage Digital Sampling Keyboard. Sounds can then be played from a MIDI keyboard, the computer keyboard, or saved as an IFF standard file. Mirage is a trademark of Ensoniq Inc.

Transform your Amiga into a professional-quality drum machine with this software package. Easier to use than hardware-based drum machines because everything is displayed graphically on screen. Enter drum patterns quickly and easily in real time with visual feedback and editing. Create realistic drum tracks with any of the 100 drum and percussion samples that are included or use your own unique IFF one-shot samples. Dynamic Drums also has full MIDI implementation and even becomes velocity sensitive when triggered from a MIDI keyboard.

A powerful MIDI sequencer that takes full advantage of the Amiga's sound, graphics, and sophisticated user-interface. Dynamic Studio is perfect for professional applications due to its sophisticated editing capabilities and SMPTE support. It is also ideal for home studios, because in addition to sequencing MIDI instruments, Dynamic Studio has a built-in drum machine, and the ability to playback instruments translated with Sound Oasis.

*new wave*  
S O F T W A R E

No Copy Protection

P.O. Box 438 St. Clair Shores, MI 48080 (313) 771-4465

### Forms Editor

The Forms Editor is a separate program, presumably so everything will fit into a 512K Amiga. You can multitask it from within SBPro, or run it separately from a Workbench icon. Initially, I found the forms editor difficult to work with, though I learned quickly. This is a purely personal reaction—a friend who is a graphic designer found it easy to learn. It is object-oriented, treating each field, piece of text, graphic, etc., as a separate object to be moved or deleted individually, though SBPro 3 lets you select a group of objects. Everything is click-and-drag, with improvements in SBPro 3, and each object has its own attributes, color, frame, etc.

A tip: In the forms editor, you select the resolution (i.e., number of colors) before you start to create the form. When I finished my first form, my 16-color graphic loaded into it as a unrecognizable 2-color mess! Very frustrating! Precision offers a fix for this in SBPro 3—you can cut and paste the entire form layout onto a new form with the correct color resolution. A form can cover multiple files, including cross-file

calculations. (Links are set up by menu and mouse.) A valuable SBPro 3 enhancement is transactional processing; it is now possible to establish a one-to-many link in a form, updating all related files without extensive programming.

A form can also now include "buttons"—calculations that execute when you click on them—recommended particularly for accessing help files on other form pages. The Forms Editor adds immensely to the power and flexibility of Superbase Professional. While the programming language can be used to create forms, it's a lot easier to do it with the mouse and menus!

### Programming language

The syntax of SBPro's DML (Database Management Language) is essentially BASIC, but with labels rather than line numbers, and many keywords that are specific to SBPro. (Most Superbase programmers have the experience of figuring out an elegant algorithm only to find a single command already exists for the purpose!)

If you've used the version of Superbase for the Commodore 64 or 128, this is similar, but with new commands for the Amiga. For example, REQUEST pops up a requester (there are 20 types), and MOUSE allows mouse input. But if you've never tried programming before, don't be put off. The syntax is different from AmigaDOS, but easy enough. Just read the manual a piece at a time and experiment. It's worth the effort.

### Text editor/Mail Merge

SBPro's text editor is perhaps its weakest feature. To be fair, it is intended only for writing merge letters and text files, not as a substitute for a word processor, and it has been enhanced slightly in Superbase 3.0. However, it could still use improvement.

The text editor opens from Superbase, in a window of its own, with separate menus. You can choose the style (bold/underline/italic) but not the font. Margins are set with the mouse, on a "ruler", but you can't change the pre-set tabs. Formatting is limited to changing margins—no justification, centering or indenting.

(continued)



The MAIL MERGE is straightforward. You create a separate file with the fields you need, and a text file with the field names indicated and the style selected. The mail merge will truncate extra spaces, and (in Superbase 3.0) close up any empty fields in an address, for example.

A separate LABEL function prints mailing labels up to four across, also closing up empty fields. I found this function easy and useful. A requester lets you define the size and spacing, and you can do a two-row formatting test as well as a single label without committing yourself to printing an entire series.

### Generating reports

Superbase Professional offers two methods of generating reports—the QUERY requester and a REPORT created with the forms editor or programming. Both can handle multi-file reports, with links established through common fields. I use the query to get quick answers for my own use, and the report form to get a polished, formatted report for the client. Both the results and the report templates can be saved to disk for future use. (The forms editor generates a program which can be further edited in the program editor.)

You create a query with mouse clicks, though you can type directly into the field name panels. You can include formulas, subtotals and totals, sort on more than one field, and select by multiple criteria (the FILTER). A SUMMARIZE option gives just the totals and subtotals. The report form adds BEFORE/AFTER GROUP/REPORT text, so you can fill out your report with more information—especially useful for summarized reports.

### Protecting your database

Passwords are available in Superbase: files can be made read-only or deletion-proof. (Passwords can be changed in Superbase Professional.) A user can also be "locked in" to your DML program with a BREAK command.

### Wishlist

I'd like the string requesters and the query function to accept text editor commands and text from function keys, and I'd like to be able to design my own requester choices. At the moment, the program limits you to specific selections. Requester 14, for example, is a list of

query files in the current directory. I'd like to see a "shortcut" way to save a record while under program control. At the moment you seem to have to cycle through all the fields. Superbase 64 let you do this with a shifted return — it seems a pity to lose such a useful function.

### Small annoyances

- Because of its complexity, Superbase Professional can be a little confusing. Most Amiga programs use the same or similar conventions for common functions, especially those using the clipboard. R-AMIGA X (cut), R-AMIGA C (copy) and R-AMIGA P (paste), for instance. SBPro uses CTRL keys instead, presumably because the AMIGA keys are used for other things. As the program evolves, keeping track of which keystrokes work in which module, including the last-minute additions in the read-me files, gets a little confusing.

- SBPro will only look at one directory—the current directory—at a time. This can be a problem if you're working with floppies. You may wish to store queries, reports, etc. on a different disk from your files, and while you can load and save from another directory, there's no way to check for correct spelling, etc. (Yes, you can still multitask a CLI, but still...)

### And one large nuisance

The query function needs a better "abort". As it is, a click on the "stop" button (or CTRL-C) stops the query only after it retrieves the first match. Too many times (usually late at night!) I have created a multi-file query and sent Superbase into total confusion by asking it to select or sort on a field from the wrong file. My mistake, of course, but a program of this quality should be able to help us out of our mistakes.

### Conclusion

Other than these few problems, I have been very happy with the program. It has allowed me to create an application, debug it enough so someone else can use it, and produce the reports I need, all with a minimum of effort, and minimal programming experience.

### Precision Inc.

8404 Sterling St., Suite A  
Irving, TX 75063  
(214) 929-4888

## Superbase 3.0 Upgrade

Several features and enhancements have been added in Superbase Professional 3.

- Telecommunications (See explanation above).

- Transactional processing—The ability to create forms with "one-to-many" links means that records in different files can be updated interactively without programming.

- ARexx compatibility—SBPro can respond to ARexx commands in other programs, run ARexx programs, or pass commands to other ARexx-compatible programs, including Precision's upcoming Superplan.

- Control panel access within programs—A WAIT PANEL statement lets you use the "VCR" controls from within a program.

- Import/export file compatibility—You can now import and export files for dBaseII and III, Logistiks and Lotus spreadsheets, and .DIF file structures.

- 'Lookup' requester—A new requester lets you validate a field with a requester that shows the closest matches to what you typed in.

- Custom heading—Tired of "Please open a file"? Now it says whatever you want.

- Cut and paste—Superbase 3.0 uses the Clipboard to cut and paste within the text editor, program editor and forms editor—even from field to field in data entry.

- Forms editor—Enhancements include outlined areas, pen and paper mode selectors, a currency gadget, test printouts, cut and paste, and a form status display.

- Keyboard equivalents—There are now keyboard equivalents for the "VCR" control panel.

- Mail merge—Empty fields are closed up. The text editor now allows the use of variables, allowing mail merge under program control.

### Revised Commands

These commands have been amended or expanded: BLANK, ENTER, EXPORT, IMPORT, OPEN FIELDS, REQUEST, FILE, INDEX, MENU, POSITION.

### New Commands

There are several new commands and functions in addition to those for communications and transactional processing. These are: LOAD SET, SELECT FORM ROW, SET EDIT, SET HEADING, SET NOW, SET TODAY, SET REQUEST, UPDATE FORM ROW, EXISTS, MOUSE, PANEL, WAIT,

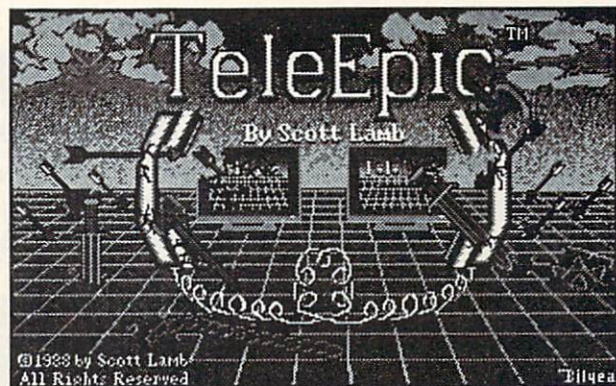
—Marion Deland



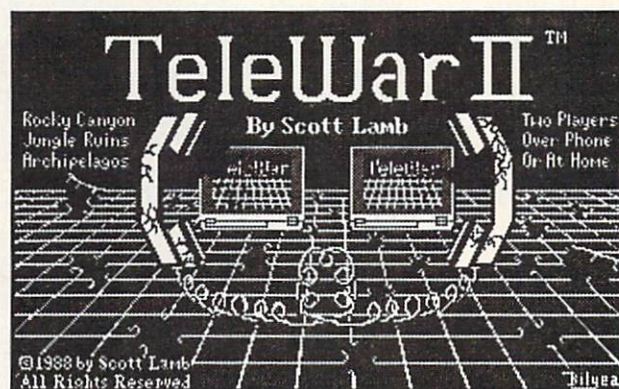
# THE ULTIMATE STRATEGY BATTLE SIMULATIONS



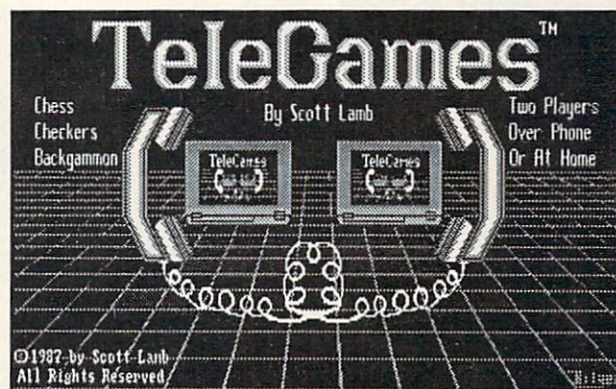
**TeleWar**, the *Ultimate Strategy War Game*, allows you to play against the *Ultimate Opponent*. Playing against an unpredictable human opponent assures you that victory is only sweet until the next game and that each TeleWar match is an exciting clash of intellect and intuition. **\$39.95**



**TeleEpic** is the *Medieval Version of the Ultimate Strategy War Game*. Instead of commanding tanks and artillery, you lead a band of heroes waging war against your mortal foe. Test your skills wielding a mighty axe or sword. Throw a dagger or check your accuracy as an archer. **\$39.95**



**TeleWar II**, the *Ultimate Strategy War Game Revisited* with new terrains and enhanced game play. Experience the ecstasy of blasting your best friend to rubble or disintegrate someone you don't even know. Watch out! You may be the next casualty peering out of a twisted pile of smoldering steel. **\$39.95**



**TeleGames**, like all our 'Tele' products, provides the enjoyment of competition without the problem of where to play. Competition is enhanced with smooth 3D perspective graphic animation and digitized sound effects. Players may compete on one computer or 'TeleConnect' via modem. **\$34.95**

## INTRUDER ALERT

and time stamping of events, external program launching, and utility support for BSR/X-10 controlling. Like all Software Terminal products direct support from the author and designer is available through our BBS service or via BIX vendor support area 'TeleGames'. **\$159.00**

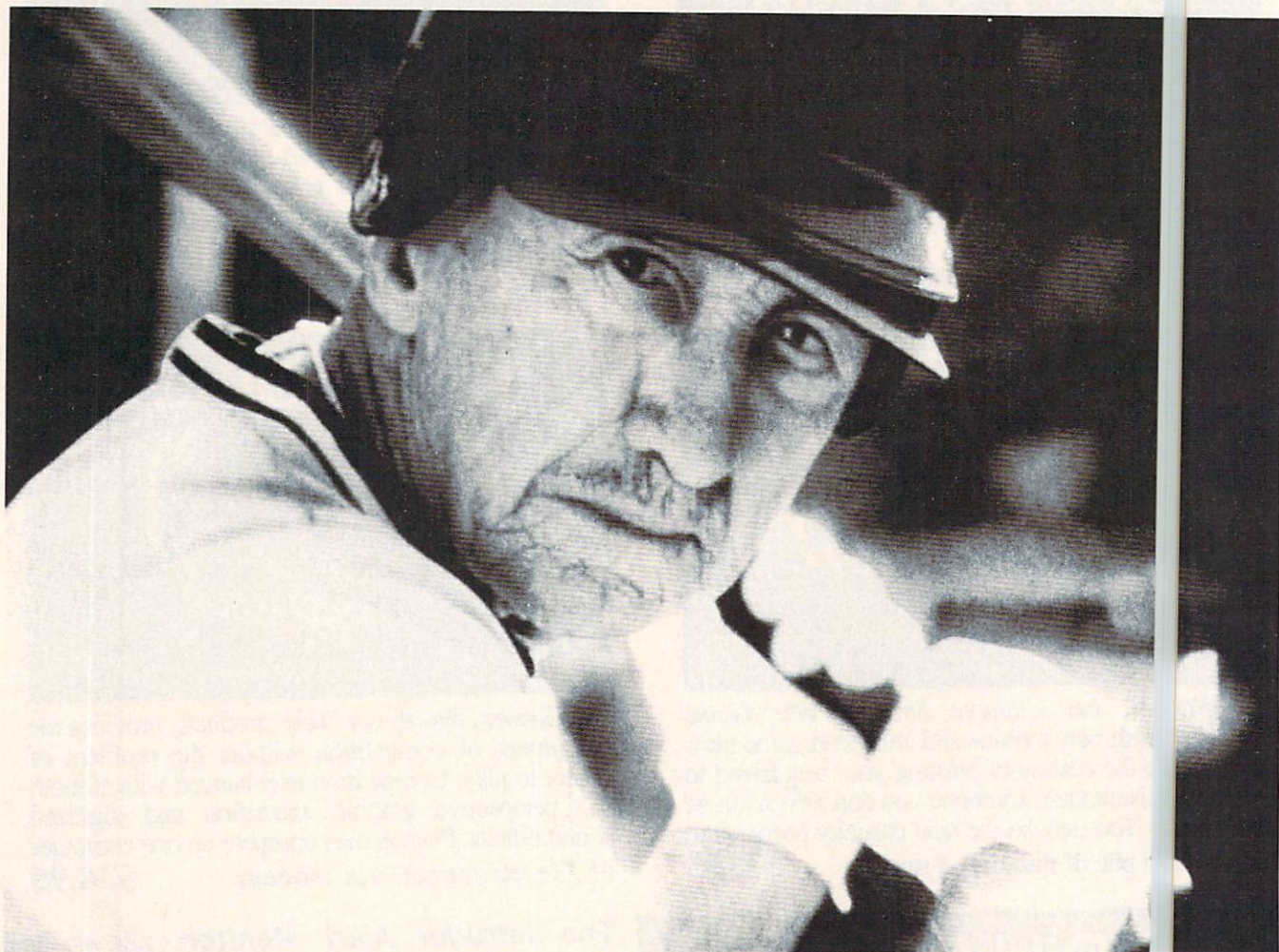
**AVAILABLE  
FROM YOUR LOCAL  
DEALER!**



3014 Alta Mere Drive  
Fort Worth, Tx 76116  
Phone: (817)-244-4150  
BBS: (817)-244-4151



# **\$500,000,000 OF RESEARCH HELPED CLIFF SHAW PLAY BASEBALL AT AGE 85.**



In November 1973, Cliff Shaw was stricken with cancer.

Fortunately, it was detected early enough. And with surgery, Cliff was able to continue living a healthy, active life.

There was a time when such a diagnosis was virtually hopeless.

But today, cancer is being beaten. Over the years, we've spent \$500,000,000 in research. And we've made great strides against many forms of cancer.

With early detection and treatment, the survival rate for colon and rectal cancer can be as high as 75%. Hodgkin's disease, as high as 74%. Breast cancer, as high as 90%.

Today, one out of two people who get cancer gets well. It's a whole new ball game.



**AMERICAN CANCER SOCIETY®**

Help us keep winning.



# MICROFICHE *Filer* PLUS

by Ronald Currier

### Introduction

Software Vision's Microfiche Filer Plus is a product of its environment. Just as Lotus 1-2-3 could not exist without the power of the IBM PC, and just as Microsoft Excel needs the power of the Macintosh, Microfiche Filer Plus could not have been written for any personal computer other than the Amiga. Its use of the Amiga's graphics, multi-tasking, and hardware capabilities set MFF+ light years ahead of the pack.

### Overview

MFF+ is a flat-file, memory-resident database. That is, only one data file may be opened and manipulated at a time and that file must fit completely in memory. While working on only a single file at one time may seem like a serious limitation, the Amiga's multi-tasking and MFF+'s ARExx support allow complex relational-like databases to be easily constructed.

MFF+ comes on a single, non-bootable, non-copy protected disk. The program itself takes up only a small portion of the disk, the remaining space is filled with sample databases and ARExx macros. Included are a simple address book, a simple picture database, a complete database of the first 128 Fish disks, a database of articles from last year's *Amazing* and *AmigaWorld*, a video storyboard, and an Invoice/Order entry system.

Microfiche Filer Plus is designed around a microfiche reader metaphor. Multiple records are presented on the screen as if they were on a microfiche card. The screen is divided into three windows: a small, fixed size window in the upper right of the screen represents the fiche, a 3/4 screen size window on the left represents the microfiche magnification, and a one quarter screen size window in the lower right corner contains the Form List (which defines how the data is shown). These three windows are normally opened on their own screen, but you may also choose to open them on the WorkBench screen.

Each record in the database is represented by a tiny white rectangle in the fiche window. A "magnifying glass" may be grabbed with the mouse and moved about the fiche window to

examine records. The records under the magnifying glass are displayed in the magnification window. Double-clicking on a record in the magnification window brings up the record editor, allowing the data in the record to be modified.

### Field types

MFF+ supports four types of fields: Text, Number, Calculated, and Picture. Text fields may contain up to 32,000 characters. One of the unique features of MFF+ is that Text fields are not a fixed length, they are always variable length. This has two advantages: you don't need to plan in advance the maximum length of your text fields, and you don't waste space in your database by choosing a overly large maximum length. Text fields are normally sorted using a dictionary

(continued)





(Top) This MFF PLUS order entry database demonstrates the power of user definable forms.

(Bottom) This MFF PLUS digitized business card database illustrates simultaneous display of text and pictures.



sort—uppercase and lowercase characters are considered the same. Instead, you may use an ASCII sort, where lowercase characters follow uppercase.

Number fields are identical to Text fields in that they are of variable length and essentially unlimited length. However, they are assumed to contain numbers and therefore sort and print differently. The content of a Number field is stored as text and is converted to a number when needed for sorting. This conversion is done by ignoring everything in the field except a minus sign, the digits 0-9, and your choice of a decimal separator (either "." or ","). This allows you to use arbitrary separators in Number fields. For instance, the dates "88/11/15" and "881115" are both sorted as 881115.00.

Number fields may have formulas associated with them, in which case they are considered Calculated fields. Formulas consist of an equation and an optional display format. Equations may be any simple arithmetic expression made up of constants, field references, parenthesis, and operators (add, subtract, multiply, and divide). The optional display format allows the number of

digits on either side of the decimal separator to be controlled. These simple calculated fields are sufficient for most applications. Fields requiring more complex equations must be set via ARexx macros (see below).

One place where MFF+ really shines is in its support for Picture fields. Microfiche Filer Plus is the only Amiga database program which can display any IFF picture, in low-res, hi-res, HAM, overscan, brushes, and icons. It is also the only database which can show pictures from multiple records simultaneously. MFF+ databases do not contain the actual picture. Rather, they contain pointers (the filenames) of the picture files. The pictures are read from disk and displayed either automatically when the database is opened, or on demand. This allows you to build a picture database that spans multiple disks.

Another unique feature of MFF+'s handling of Picture fields is that what you see in the magnification and edit windows is not the actual picture, but a squeezed and translated version. This allows multiple pictures of any size, resolution, and format to be displayed simultaneously in the same window.

Several options are available to control the squeezing and color translating functions. Once the options are set, MFF+ automatically resizes and recolors the pictures. A lot of work has obviously gone into this portion of the program. While this process takes some noticeable time, MFF+ performs some very heavy-duty image processing in a fraction of the time required by most of the programs available. And lest you think MFF+ only displays squeezed and translated pictures, the original picture is only a mouse-click or keystroke away.

## Forms

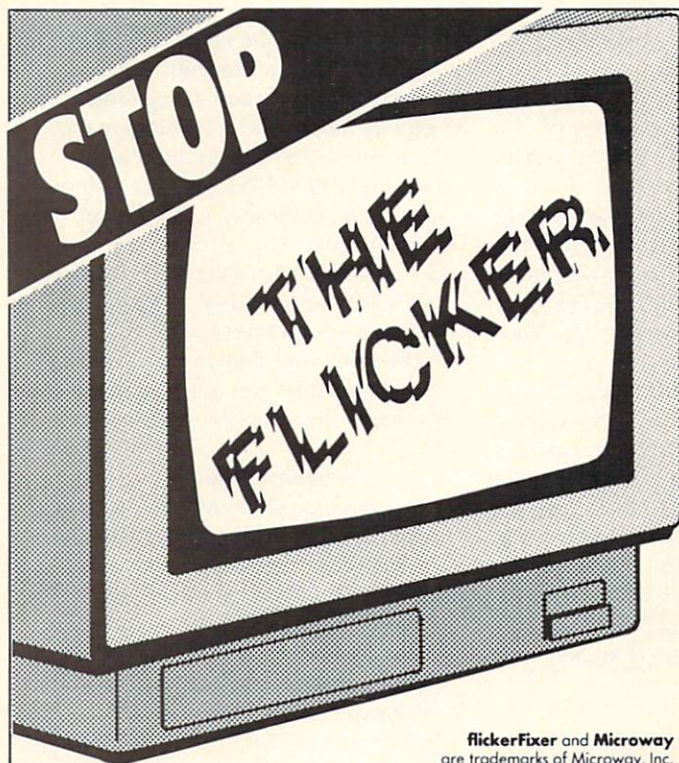
MFF+ is a forms-oriented database. That is, all display, sorting, and printing of data is done via one or more forms. A form is a template which describes how data. Forms are created and edited by double-clicking on any of the forms in the Form List window. This brings up the Form Editor, which allows you to define which fields of your database are to be displayed as well as their size and position within the display window. Except for entering descriptive text into the form, all form editing is done using the mouse. This process requires a fair amount of dexterity, as some of the gadgets are very small. However, the whole process is so elegant and simple that the skill required to hit the gadgets on the fly is easily forgiven.

Within the Form List window are eight form "slots", with each slot serving a different purpose. For instance, the form contained in the Display form slot controls how the records are displayed in the Magnification window. The Sort form slot controls sorting and so on. For the record, the eight slots are Print Title, three Print forms, Print Totals, Sort, Edit, and Display. Any number of forms may be created. Those forms not assigned to a slot are shown at the bottom of the Form List. Note that a form may be assigned to multiple form slots. Forms are assigned to a slot by simply dragging the form name into the appropriate slot.

## Sorting

An in-memory database sorts faster than a disk based database. In the case of MFF+, "faster" doesn't begin to describe the sorting speed. MFF+ performs a two key sort of 850 records in less than four seconds. Because of its speed, MFF+ eliminates one of the major





**flickerFixer and Microway**  
are trademarks of Microway, Inc.  
Amiga is a registered trademark of Commodore.  
Multisync is a registered trademark of NEC.

# Microway

P.O. Box 79  
Kingston, Mass. 02364 USA  
(508) 746-7341

32 High St.,  
Kingston-Upon-Thames, U.K.,  
01-541-5466

58 Atchison St.,  
St. Leonards, NSW, Australia  
02-439-8400

**flickerFixer** eliminates your Amiga 2000's interlace flicker and visible scan lines. The result: **superior quality color or monochrome graphics and text** — for a full range of demanding applications, including CAD, desktop presentation, graphics, animation, and video.

Reviews are impressive: Commodore Magazine 12/88: **Best of 1988 Award**. AMIGAWORLD 12/88: **#1 Readers' Choice Award**. Amiga GURU 5/88: "The display is fantastic . . . It is the best display we have ever seen on any computer system."

**flickerFixer** fits into the Amiga video slot, is fully compatible with all user software, and does not modify the standard Amiga video signals. The board upgrades the Amiga 2000 with a flicker free 4096 color palette, has an overscan mode that features a screen size of 704 x 470 pixels and drives most of the popular PC Multisync and VGA monitors, including the NEC Multisync and Zenith ZCM1490.

## flickerFixer™

**Advanced Graphics Adapter For The AMIGA® 2000**

**flickerFixer** is priced at \$595. It is made in the USA and is FCC Class B approved. For more information or to order, call Microway Sales at (508) 746-7341 or your Amiga Dealer.

problem areas with conventional database programs: Index files. Index files were invented to speed sorting and searching of large databases; a task they perform very well. Unfortunately, index files require that you decide (at database creation time!) which fields are to be used as keys. If you ever need to sort on another field, you must change your database definition. MFF+ allows you to sort your database on any combination of fields and to change the sort at any time.

MFF+ sorts by defining a Sort Form which describes the fields to sort on and picking SORT from the menus. The screen is automatically updated to reflect the new ordering. New records are automatically inserted in the correct sort order.

Several options control the sorting process. Sorts may be done in either ascending or descending order, text fields may be either case sensitive or insensitive, and automatic sorting of new records may be disabled. This last option suppresses the constant screen refresh caused by sorting when adding large numbers of records.

### Record Selection

In addition to extremely fast sorting, an in-memory data base also gives you quick record selection (also referred to as database searching). MFF+ allows any combination of fields in the record to act as search keys. When the Selection editor is opened, a list of field names and selection criteria is presented. Enter the desired selection criteria, click on the close gadget, and the requested records will be selected. Since the selected records may be scattered throughout the database, a menu option moves all of the selected records to the front of the display.

MFF+ provides a rich set of selection criteria. Fields may be examined for exact, greater than, and less than matches (text and numbers) or beginning, containing, and ending text matches. A full set of Boolean (AND, OR, NOT) operators are also available for use within a field. One slight deficiency (compared to some other database programs) is that multi-field criteria are always connected with AND. That is, you can't select all records based on field 1 OR field 2. However, once again, a

simple AReXX macro can be written to perform any selection you can dream up. This restriction—supporting only AND operations between fields—was likely done to simplify what can be an extremely complex function. The current "standard" for selecting records from a mainframe database is SQL (Structured Query Language), which has a highly complex programming language of its own.

### Printing

Printing is controlled through a combination of the Print Options Editor and five forms: Print Title, Print Totals, and Print Forms 1 thru 3. The Print Options Editor allows you to control which page the title prints on, which combination of Print forms 1, 2, and 3 are used, the number of columns and records per page, whether totals should be printed, and the number of copies desired. Another sends the report to a file rather than to the printer.

If your Print Forms contain no Picture fields, MFF+ prints the report in text mode. However, if your form contains Picture fields, MFF+ will print the entire report (including the text) in graphics mode.

(continued)



There are two limitations to the type of reports that MFF+ can print. First, MFF+ provides no support for printer options such as bold or underline. This can be overcome by imbedding printer control commands in the form in appropriate places. Second, because of the way forms work, all fields are printed with fixed lengths. This means two text fields cannot be printed next to each other. This yields somewhat ugly mailing labels. Again, ARexx comes to the rescue. With ARexx's powerful string processing functions, a dozen lines of ARexx code will produce beautiful mailing labels.

### Performance

Several internal changes have been made to Microfiche Filer to improve its performance. Software Visions claims that MFF+ loads databases twice as fast as the original MFF and redraws the screen up to ten times faster. While I didn't have a copy of the original to compare against, table 1 lists several timing measurements.

**Table 1**

	Load	Insert	Sort	Select	Update
100 Records	47/sec	15/sec	384/sec	100/sec	9/sec
1000 Records	79/sec	10/sec	252/sec	55/sec	9/sec

In addition to speed improvements, MFF+ uses a new internal memory allocation scheme. Even with an increase in the size of the program file, MFF+ allows you to have more records in memory than the earlier version. Over 100 address book-type entries may be stored in a half megabyte of memory. On a one megabyte machine, this increases to several thousand.

### Macro Programming

The power of MFF+ can be enhanced with the addition of the ARexx macro language. For those unfamiliar with ARexx, it is a programming language similar to BASIC or PASCAL. ARexx is not included with MFF+, it must be purchased separately.

Nine macros are available for each database. All of the data manipulation functions of MFF+ are available through ARexx, including adding and selecting records, displaying pictures, and changing form lists. None of the definition functions are available

however, so you can't change the definition of a record from within a macro.

With ARexx macros you can automate repetitive tasks like convert all records to uppercase, perform sophisticated searches and selections, or print fancy mailing labels. MFF+ includes a pair of macros to import and export databases to and from field delimited files (such as those created by MaxiPlan and ProWrite).

In addition to using ARexx as a powerful macro language, MFF+ can also act as a server in an ARexx environment, so not only may ARexx macros be called from within MFF+, but MFF+ functions may also be performed from within other ARexx capable programs. For instance, a mail merge macro could be written to take the body text from TxEt Plus and the names and addresses from MFF+.

### Applications

Over the last 4 months, I've been using MFF+ extensively for two different projects. The first is a simple database of

changes made to the ten modules of a software product I am developing. Each record contains the product revision number,

problem and resolution descriptions, and the modules affected. It is now a simple task to search the database to find which modules were changed in version 12 or to find how many times I have fixed the "Memory List Corrupt" error.

While the first database I built was simple and could actually have been maintained with a text editor, the second uses the full power of MFF+. I combined MFF+, ARexx, a laser disk player, a genlock, a framegrabber, and a video controller to build a video database. The database catalogs each edit point within a movie on video disk. Each record contains scene, disk, and chapter numbers; scene and chapter names; location of the scene; characters in the scene; beginning and ending video frame numbers of the scene; the running time of the scene; and a picture of the first frame of the scene. The running time field is computed from the starting and ending frame numbers using a calculated field.

Once the database was built, another ARexx macro was written to play

the scene from the video disk. When the macro is run, the currently selected records are read, and the starting and ending frames and total running time is computed. The entire macro is approximately 125 lines long, half of which are comments and constant assignments.

This is what puts MFF+ head and shoulders above other currently available database programs (Amiga otherwise): the ability to use ARexx to combine MFF+ and other programs into a nearly seamless environment.

### The Down Side

The requirement that the entire database fit in memory may limit MFF+'s use in some cases. If your database requires hundreds of thousands of large records, MFF+ may not be appropriate. (But then, an Amiga may not be either!) However, most real world applications will easily fit on a one megabyte machine and many will fit on a 512K. With one megabyte becoming the norm, and three or more megabyte not uncommon, the benefits gained by a memory resident database outweigh the negatives.

In their zeal to provide keyboard shortcuts for as many menu functions as possible, Software Visions stepped on the standard string request shortcuts (RightAmiga-Q and RightAmiga-X). The Escape key is provided to perform the same function as RightAmiga-X (clear string). While this is certainly not a major problem, it is annoying when switching between MFF+ and other applications.

### Conclusions

If you have data that needs to be organized, Microfiche Filer Plus deserves a look. If you don't consider yourself a programmer, nothing is simpler and more powerful to use. If you are willing to do a little programming, the combination of Microfiche Filer Plus and ARexx leaves the competition in the dust.

**MICROFICHE Filer Plus** \$179

**Software Visions**

P.O. Box 339  
Framingham, MA 01701  
(800) 527-7000  
(508) 875-1233



# AmiFORUM

Orlando, Florida

by Stephen Pietrowicz

On January 14th and 15th the first regional Amiga show, AmiFORUM, was held at the Hyatt Regency in Orlando, Florida. AmiFORUM is a "mini" version of the larger AmiEXPO shows which are held throughout the year. Twenty-two different companies spent the weekend giving Floridians a chance to look at a variety of different Amiga products.

ASDG's Perry Kivolowitz was on hand showing off ASDG's multi-serial port adaptor card which works in conjunction with the previously released TWIN-X board. Perry demonstrated the speed of the card by running a file transfer between two Amigas running at 19200 baud, and a bar code reading application at the same time. ASDG was also drawing quite a bit of attention showing off their previously released professional color scanner. They had a very large color poster in their booth they scanned using a color post card with their Scanlab software and color scanner. The resulting color printout was pretty impressive!

Are you a video professional in need of a collection of fonts? Arock Software has the answer: The Masterpiece Professional Font Collection. Arock brings you a high quality collection of 110 different typefaces, not tired old converted Macintosh fonts. The folks at the booth told me that if you call or write, they will send you a free demo disk to view the different typefaces they offer. See the exhibitor listing at the end of this article.

The folks at Incognito Software showed off some of their recently released, and soon to be released, products. While one Amiga was showing off some of Incognito's arcade games like Targis and Vyper, show attendees were given sneak previews of

more serious applications. Incognito told me they would be coming out with Opticks, a new ray tracing program, by February 1st. Many BBS systems offer ANSI graphics as an option when you log into them. Incognito's Atredes BBS, offers Amiga graphics to people that dial into it. They expect a well known Shareware terminal program would support the Atredes BBS graphics protocols, so anyone will be able to get the full effect of the system. Also, in the "alpha" stages is a program called "Kingdoms of England", which was described as "a cross between Defender of the Crown, Faery Tale Adventure, and Fire Power".

Synthesis was on hand showing off amazing 3D ray traces of words made with their new product, InterFont. InterFont allows you to create your own fonts by tracing around letters by using their InterFont designer program. Once you've gotten the letters traced, just type in the words you want, and you can

convert them to any format that InterChange supports, including Sculpt-3D, VideoScape, and Professional Page. Synthesis plans to have new modules for VideoScape 2.0, Sculpt-Animate (with heirarchy support), Turbo Silver 3.0, and the new Pro Draw format.

A truly unique sight was at the RGB Video Creations booth: A rack-mounted Amiga! RGB has rack-mounted an Amiga 2000 for video applications, and was showing it off with their new AmigaLink software when controls video devices in their video local area network.

#### *Some other points of interest:*

One area of the show was dedicated to demonstrations, giving different vendors a chance to show off their software and hardware. I think this concept was a very good idea, and I hope the AmiEXPO folks continue to do this in future shows. Particularly interesting was an instructional video

*(continued)*



*This first regional "mini AmiEXPO" show was a success.*



cassette on how to use your Byte-by-Byte ray tracing software more effectively. (The cassette is available from Byte-by-Byte).

- Andre Frech and other Commodore representatives were on hand to answer questions about current and future Amiga products.

- Three hour "Master classes" were given by:

#### Jim Sachs

Amiga Graphics and Artist's Techniques

Steve Segal - Amiga Animation

Cal Vornberger - Amiga Video

#### Eric Lavitsky

Programming the Amiga in "C"

Each was split into separate beginner and advanced classes.

- Creative Computers had a booth that was packed with hardware and software (and people!). Many happy people walked away with greatly discounted Amiga products.

- Lattice was showing off the features of Version 5.0 of their C compiler to show attendees, as well as showing off their new source level debugger.

- Impulse was selling its Turbo Silver 3.0, Terrain, and new Data and Objects disks. They also had a camera set up showing off its new VD-1 system, a combination Video Digitizer, Frame Buffer and Frame Grabber.

- VidTech International was showing off their broadcast quality "VidTech Scanlock" genlock.

- DigiTek Software showed off some new software releases: PowerStyx (a variation on the QIX arcade game), Western Games, and a new course disk for their popular "Hole-In-One Miniature Golf".

- I was told that Joe Lowy was on his way to Europe to finalize plans for an AmiEXPO show in Germany! There are many European developers that are very interested in it, and by the looks of some of the products in the European magazines, it should be interesting! Look for more information on this show soon!

All in all, the show coordinators, exhibitors, and attendees seemed very pleased with this first regional "mini AmiEXPO" show. It was a good chance for developers to get away from the winter weather and escape to the Florida sunshine.

•AC•

## AmiFORUM Exhibitors

AMI COMP CENTER  
87 Alafaya Woods Blvd.  
Oviedo, FL 32765  
407-366-2000

AMUSE, New York Amiga User's Group  
151 1st Avenue, Suite 182  
New York, NY 10003  
212-460-8067

ASDG, Inc.  
925 Stewart Street  
Madison, WI 53713  
608-273-6585

Arock Software  
1306 East Sunshine  
Springfield, MO 65804  
800-288-2765

Byte by Byte  
9442 Capitol of Texas Highway  
Austin, TX 78759  
512-343-4357

C Ltd.  
723 East Skinner  
Wichita, KS 67211  
316-267-3807

Commodore Business Machines, Inc.  
1200 Wilson Drive  
West Chester, PA 19380  
215-431-9100

Computer Specialties  
7705 Technology Drive  
Melbourne, FL 32904  
407-725-6574

COMPUTERS, ETC!  
4521-A Bee Ridge Rd.  
Sarasota, FL 34233  
813-377-1121

Creative Computers  
318 Wilshire Boulevard  
Santa Monica, CA 90401  
213-394-7779  
800-872-8882

Digital Dynamics  
739 Navy St.  
Santa Monica, CA 90405  
213-396-9771

DigiTek  
164 W. Seneca, Suite 4  
Tampa, FL 33612  
813-933-8023

Emerald Intelligence  
334 South State Street  
Ann Arbor, MI 48104  
313-663-8757

Fuller Computer Systems  
P.O. 9222  
Mesa, AZ 85204-0430  
602-835-5018

Great Valley Products  
225 Plank Road  
Paoli, PA 19301  
215-889-9411

IMPULSE  
6870 Shingle Creek Parkway  
Suite 112  
Minneapolis, MN 55430  
612-566-0221

Incognito Software  
34518 Warren, Suite 149  
Westland, MI 48185  
313-462-2148

Lattice, Inc.  
2500 S. Highland Ave  
Lombard, IL 60148  
312-916-1600

RGB Computer & Video Creations  
3944 Florida Blvd, Suite 10  
Palm Beach Gardens, FL 33410  
407-622-0138

Sun Coast Amiga Club  
2931 South Pines Drive #d  
Largo, FL 34641  
813-539-1257

Syndesis  
20 West St.  
Wilmington, MA 01887  
508-657-5585

VidTech International  
2822 N.W. 79th Ave  
Miami, FL 33122  
305-477-2228



# Super Bug Bytes

## The Bugs & Upgrades Column

A recent notice left on PeopleLink prompted me to verify a bug in the New Century Schoolbook font that comes with Professional Page. On-screen display of the font is correct, but when Bold is selected, the printer prints Bold-Italic, and vice versa. Syn\*Ed of PeopleLink suggests making the following correction to a BACKUP of your font files.

Using NewZap, make the following changes to the file

"Fonts:NewCentury.metric"

At block 1, location 004, change  
longword: old=00000CA0 ->  
new=0000268C

At block 1, location 00C, change  
longword: old=0000268C ->  
new=00000CA0

At block 7, location 106, change word:  
old=0002 ->  
new=0004

At block 20, location 0F2, change word:  
old=0004 ->  
new=0002

All numbers listed above are hexadecimal except for block numbers, which are decimal; the first block in a file is block 1, not block 0.

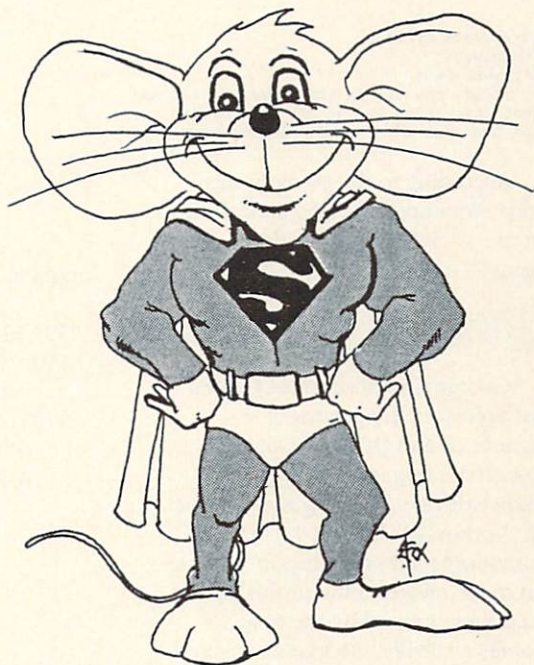
According to the posting, "This bug fix has not been blessed in any way by Gold Disk. It works just fine for me - it should work for you. I will, of course, inform Gold Disk of the problem so they can fix it in their next release of Pro Page. Ed Scherer, Plink ID: Syn\*Ed"

With the release of Workbench 1.3, I have been seeing a lot of "bug reports". Some of them really do seem to be bugs; but some turn out to be incorrect installation procedures. Commodore has caused a couple of problems with a typographical error in the latest pre-release version of Workbench 1.3. The release version documentation has the correct information, which revolves around the MaxTransfer entry in the mountlist. The number following the MaxTransfer keyword is the number of blocks, not the number of bytes.

Though this problem does not affect the vast majority of Amiga users who will be buying the Workbench 1.3, developers may have run into problems with their software as they made their own changeover to version 1.3.

There have been many postings regarding the use of FastMemFirst 1.3 with Kickstart 1.3. Many individuals are receiving random Gurus upon startup and several other severe problems that mysteriously clear up if FastMemFirst is either omitted or replaced by FastMemFirst 1.2. If you can shed some light on this problem, please contact me via Amazing Computing, or via People Link or Compuserve.

The A2090 and A2090A controllers from Commodore automatically mount the first hard drive in the system.



Therefore, until Workbench 1.4 allows the Fast File System to be installed on automatically mounted drives, format DH0: —the first partition on the hard disk—with the original file system. This limitation has caused people who use the A2090 card to format a very small partition to the old system, then make a large dh1: partition which can be formatted and mounted in fast file system format.

R\*Staples left a posting on People Link that provides a workaround. I have not tried this solution, but comments left on the net indicated that others have. To implement the procedure, you will need to obtain ASSIGNDEV, a public domain program that allows the reassignment of devices. ASSIGNDEV can be found in most of the information service Amiga section libraries. Before starting the procedure below, be sure to backup your hard disk, as the Prep program will completely erase information stored on your hard drive.

(continued)



Here is the procedure. Prep your drive according to the 2090 documentation, and set up the mount according to the fast file example on the 1.3 Workbench mountlist and mount that device.

Then put this in your startup-sequence:

```
system/FastMemFirst
BindDrivers
ASSIGNDEV DH0:
;tells the system that there is no DH0:
Mount dh0: ;this is the new FFS section
;put the rest of your startup-sequence here
```

According to the posting, this startup sequence has been used with both the ST-506 Miniscribe drive and the Seagate ST-157N.

George Chamberlain of Central Coast Software, the creator of Quarterback and DOS-2-DOS, has discovered a bug in the AmigaDOS trackdisk.device. The bug is present in both Workbench 1.2 and 1.3. The Commodore technical support staff has been made aware of the problem, and should have it fixed by the time 1.4 becomes available. George discovered the problem when he was working with an upgrade to Quarterback. The problem occurs in the time delay routine that steps a floppy drive motor, which causes the operating system to sense whether a disk has been inserted in the drive. This bug causes the random software lockups that seem to occur for no reason. The problem appears when the time delay step routine incorrectly releases a different floppy drive that should still be busy. Trackdisk.device then waits patiently to regain access to the drive, which will never happen. This locks up the task currently using trackdisk.device. The workaround is to keep a disk in all empty floppy drives at all times.

While on the subject of Central Coast Software, the upgrade to Quarterback I mentioned earlier has been finished. Quarterback 2.0 has several new features, including support for streaming tape backup units, high density 10.7 MB floppy drives, Bernoulli drives, and any drive unit that can be mounted. Quarterback has always supported a dual floppy backup, and now version 2.0 contains a dual floppy restore, speeding up both backup and restore operations. While he was at it,

George built a workaround into Quarterback that avoids the bug in trackdisk.device, should you forget to keep a disk in all empty drives.

To obtain your upgrade, mail your original disk and \$15.00 to:

#### **Central Coast Software**

268 Bowie Drive  
Los Osos, CA 93402  
(805) 528-4906

If you own Lattice C, you can upgrade to the 5.0 version for:

\$75.00 if you have 4.0 or later  
\$100.00 if you have any previous version of the Lattice compiler  
\$150.00 Plus manuals and distribution disks if you have Manx 3.4 or later.

#### **Lattice, Inc.**

2500 S. Highland Ave  
Lombard, IL 60148  
(800) 533-3577

C Ltd. has improved its hard drive utilities to support the advanced features of Workbench 1.3. SCSIDOS 3.0 was scheduled to ship in mid November. The upgrade is free via C Ltd's BBS, with limited Documentation; or \$20.00 if you order it with the printed manual. This two disk set includes a manual and the complete set of SCSIDOS tools and Utilities. To get your upgrade, order Item # 3.0 SCSIDOS.

#### **C Ltd**

723 E. Skinner  
Wichita, KS 67211  
(316) 267-3807

If you are a registered user of B.A.D., the floppy disk accelerator, you may send your original disk or \$5.00 (for a new disk), and receive the latest version of B.A.D., which supports the Fast File System.

#### **Mark Hellman**

P.O. Box 1112  
Wheatridge, CO 80034-1112  
(303) 467-1718

Amigo Business Computers has upgraded KickWork from 1.2 to 1.3, Kickwork is for Amiga 1000 owners who would like to boot from a single disk, instead of first booting Kickstart, and then Workbench. Kickwork has been rewritten to work with Workbench 1.3. If you own KickWork and have registered it with Amigo, you should have gotten an upgrade letter in the mail. Contact Amigo Business Computers for upgrade details.

#### **Amigo Business Computers**

192 Laurel Road  
E. Northport NY 11731  
(516) 757-7334

Jim Fiore of dissidents Software has written regarding an upgrade to their loudspeaker CAD program, SpeakerSim. Version 1.1 is now shipping, and should have been sent to all registered 1.0 owners. The latest version includes bug fixes, easier to use requesters, keyboard shortcuts, and enhanced error trapping. Support for the Epson Hi-80 plotter has also been provided. If you have not registered your package, you may contact them directly at:

#### **dissidents**

730 Dawes Ave  
Utica, NY 13502  
(315) 797-0343

Reports that the latest version of Deluxe Print II has a bug in the print utility are not true. According to Technical Support at Electronic Arts, you must either use version 1.2 of the printer driver software. One Deluxe Print II user suggested that you could copy the appropriate 1.2 driver to your devs/printers subdirectory of the 1.3 Workbench disk, and use Preferences to select the 1.2 driver when printing with Deluxe Print II.



# HIGHER PERFORMANCE...AND CHEAPER TO BOOT!

**FData-10** Single 3.5" External Drive ..... **\$149.95**

**FData-20** Dual 3.5" External Drive w/Power Supply ..... **\$299.95**

- Fully 1010 Compatible
- Ultra Compact
- Acoustically Quiet
- Amiga® Color Coordinated
- Daisy Chainable
- Extra Long Cable
- High Performance
- Super Low Price

**FLEXIBLE DATA**  
  
**SYSTEMS, INC.**  
 10503 FOREST LANE • SUITE 148 • DALLAS, TX 75243  
 FAX: 214-669-0021

POLICY: Shipping and handling extra. Personal and company checks require 3 weeks to clear. For faster delivery, use your credit card or send cashier's check or bank money order. Credit cards are not charged until we ship. All prices are U.S.A. prices and are subject to change, and all items are subject to availability. These prices reflect a 5% cash discount. For all credit card purchases there will be an additional 5% charge. Defective software will be replaced with the same item only. All sales are final and returned shipments are subject to a restocking fee.

**214-669-3999**

Amiga® is a registered trademark of Commodore-Amiga, Inc.



Another Deluxe Print user had loaded the new fonts on the 1.3 Extras disk into his hard disk fonts directory, and when he tried to load a greeting card done earlier, Deluxe Print locked up. He reports that it seems to load only the first 11 fonts it finds into memory, and could not read the topaz font which the greeting card used. I have not had a chance to personally verify this report, but there are several programs in the Amiga software community that can only access a limited font directory. Software developers should make font requesters that allow large font directories. Assigning fonts to various subdirectories is currently the only solution to this problem.

Dave Haynie of Commodore has released version 1.3 of DiskSalv, a public domain utility that can read a defective Amiga disk and recover as much information from the disk as is possible, allowing you to restore it to another disk. This version was released shortly after version 1.2, when it was discovered that version 1.2 has a bug that declares that almost any disk, except the RAM disk, is full, even though it isn't. DiskSalv can be found in the Fred Fish collection, or on most of the major information services.

Comic Setter from Gold Disk has a problem with occasional trashing of graphics within a Comic Setter panel. A trashed graphic appears as random colors and blocks. The graphic must be

cut, and then reloaded from disk, to correct it. Gold Disk is aware of the problem and is working on a fix for it. The bug seems to appear randomly, and can cause the user to have to redo an entire panel, on occasion. Until a fix is found, you should check each panel for proper color quality, and save your comic file each time you finish a panel.

Professional Page, also by Gold Disk has a sporadic and rare bug that crops up on occasion. If you try to import a graphic and it brings up a requester that reports "Can't find bitmap", you can work around the problem by using the mop tool to "clean out" the box. Re-import the graphic again, and it should work properly.

## Gold Disk

Box 789 Streetsville  
 Mississauga, Ontario, CN L5M 2C2  
 (800) 387-8192

While this item is not really a bug report, it is a workaround for what I perceive to be a problem regarding Rocket Ranger from CinemaWare. Rocket Ranger is the first of CinemaWare's programs to use their new high speed floppy file system. This system greatly enhances the performance of Amiga floppy drives. Unfortunately, they are shutting themselves off from single drive users, and hard disk users (many of which have only a single drive), because the game cannot be transferred to a hard disk. My own system has 3 MB of RAM, and a hard disk, but no second floppy drive. I had not bought the game

because I could not copy the second disk to RAM, or my hard disk. Now that Workbench 1.3 is installed on my system, I can make a special startup disk that creates an 80 cylinder RAD, and then use diskcopy to copy Rocket Rangers' disk 2 to RAM. I still have to run disk 1 from the floppy drive, forsaking the speed of the new Amiga Fast File System. CinemaWare should realize they are losing sales with their new format, and should consider an upgrade for those with hard disks and lots of RAM, so that they too may play the game without resorting to the purchase of external floppy drives.

The Investor's Advantage has been upgraded to version 2.0. The upgrade includes a more flexible user interface, and zoom capability, among other features. Registered owners can upgrade for \$25.00 plus \$2.40 postage. If you purchased version 1 after May 1, 1988, the upgrade is free, if you include a dated receipt or invoice.

## Software Advantage Consulting Corporation

37346 Charter Oaks Blvd  
 Mt. Clements, MI 48403  
 (313) 463-4995

(continued)



Calligrapher, by InterActive Softworks has been upgraded to autoconfigure for PAL or NTSC, and FontMover, a program that allows the easier arrangement of Fonts has also been included in version 1.05. The upgrade fee for registered version 1.0 owners is \$29.95.

#### **Interactive Softworks**

2521 South Vista Way Suite 254  
Carlsbad, CA 92008  
(619) 434-5327

A-Talk III from Oxixi, Inc. has been upgraded to version 1.0c. Enhancements include WXmodem, a protocol gaining popularity because it speeds Xmodem transfers through packet switching networks such as Tymnet and Telenet. WXmodem transfers save downloading time and money by optimizing block transfers through the network. A-Talk III also has other new features including Ymodem-G, and ZMODEM Recover/Resume of an interrupted file transfer both when invoked by the host and when selected locally on the Amiga.

A bug has been fixed in A-Talk III 1.0 concerning the DIAL command of the script language. The manual says that the parameter of the DIAL command is a string between double quotes("). In reality, the DIAL command only accepted strings WITHOUT quotes. In A-Talk III rel 1.0c, the DIAL command works as described in the manual. Therefore, from this version on, the double quotes are required.

Owners of A-Talk Plus can upgrade to A-Talk III for \$25.00 plus \$4.00 for shipping and sales tax for California residents. Owners of Maxicom can upgrade for \$50.00 plus the above listed shipping and tax. If you own Diga! or Online, you can trade them in for A-Talk III for \$60.00 plus the above charges, and the original manual cover and disk for the other products.

With regard to another Oxixi product, I received a letter from Glenn Smith of Tilbury, Ontario reporting a bug in Maxiplan 500. According to his letter, when trying to create an amortization schedule that was three columns wide, he found that all calculations were performed correctly until the program reached the 50th row. At that point, the

program crashed. He had created formulas in row 2 and used the F8 key to fill down.

He also reports that on version 1.8, the program did not crash, but no calculations were performed on the second column after row 49. If you have had similar problems with Maxiplan 500, let me know. We will see if we can verify a repeatable bug.

#### **Oxixi, Inc.**

Box 90309  
Long Beach, CA 90809-0309  
(213) 427-1227

Charlie Heath of Microsmiths has a warning regarding the use of BlitzDisk V1.0 on an FFS formatted hard disk. According to Mr. Heath, The currently released version of BlitzDisk is not compatible with FFS, and if you use it you may wind up with unexpected results when you save files. An FFS-compatible version of BlitzDisk is in the testing process, and should be available as you are reading this.

The first release version of Supra's mount command that works with the Fast File system (version 5.1a), has a compatibility problem with Central Coast's Quarterback hard disk back up utility. Both Supra and Central Coast software are aware of the problem, and Supra should have an upgrade to version 5.1a as this is being read. What is interesting to me is that the problem doesn't affect everybody. I know personally of two systems that have the version 5.1a SupraMount which are working properly with Quarterback. Caution should be used, however, as a problem with backup disks on a large hard drive filled with important data can be a catastrophe waiting to happen.

#### **Supra Corp**

1133 Commercial Way  
Albany, OR 97321  
(800) 727-8772

If you are using ProWrite, and an HP DeskJet printer, and you are having problems with the documents you are printing from ProWrite on an HP

DeskJet, New Horizons software has a fix. There seems to be a bug in the DeskJet driver (the final version as released on the Workbench 1.3 package) that sometimes causes ProWrite documents to print the last 1/6 of an inch on the next page (this will usually show itself as simply getting a blank page between each regular page). They now have a version of ProWrite that tries to work around this problem to produce correct results. If you would like this new version, just send your program disk and a letter stating that you would like the latest version that fixes this problem. There is no charge for the fix.

Note that this new version only corrects the problems with the HP DeskJet. There are no new features.

#### **New Horizons Software, Inc.**

PO Box 43167  
Austin, Texas 78745  
(512) 328-6650

If you have been using ARP with version 1.2 Workbench, and you have converted to version 1.3, you can still use most of the ARP commands. You must, however, use MOUNT, RESIDENT and PROTECT from the version 1.3 Enhancer, rather than the ARP versions of these commands. The ARP versions are not compatible with 1.3 Workbench. My personal solution to the problem of deciding which commands to use, is to use both. The only reason I have found to use ARP (besides the fact that they are slightly smaller and faster), is that they have a more useful and powerful wildcard format. Those commands that I often like to use with the wildcard structure include RENAME and COPY. I have renamed the ones I use with an A in front of the command (e.g. ARENAME and ACOPY.) If I wish to use an ARP style command, I simply preface it with A. The version 1.3 ARP command set will be released soon.

That's all for this month. If you have any workarounds or bugs to report, or if you know of any upgrades to commercial software, you may notify me by writing to:

#### **John Steiner**

c/o Amazing Computing  
Box 869  
Fall River, MA 02722



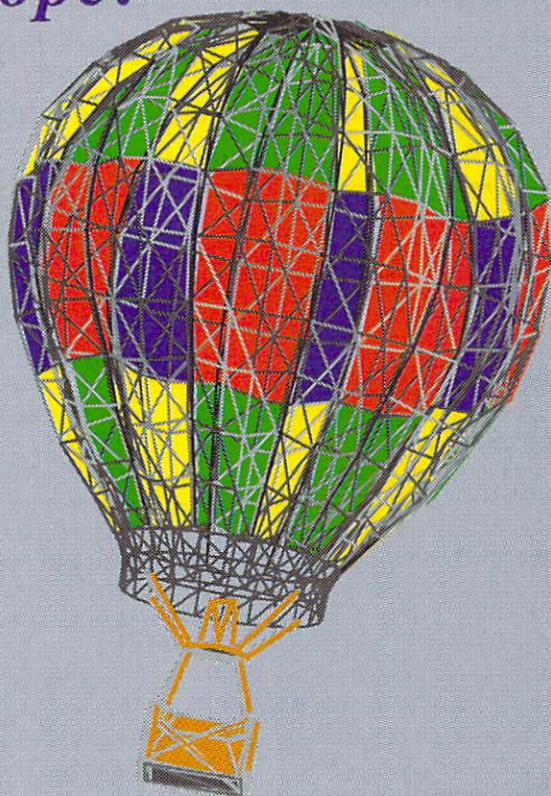
# COME ABOARD **AmiEXPO**

*and We'll Fly You to Europe!\**

The **Amiga Exposition**  
**March 3 - 5, 1989**  
**Marriott Marquis Hotel**  
**New York, NY**

**10,000 Attendees and**  
**120 Amiga Companies**  
**Will Be There.**

**DON'T MISS THE**  
**FLIGHT!**



*Admission includes the Exhibition, Seminars, Keynotes & Amiga Artists Theatre!*  
*120 Amiga Exhibitors Featuring State of the Art Software and Hardware, at the lowest prices!*  
*Master Classes Available in Amiga Graphics, Video, Programming, Animation, Music and Publishing!*  
*Seating for Master Classes is limited; call for schedule and availability before registering.*  
*For Hotel Reservations Call the Marriott Marquis at (212) 398-1900. Deadline is February 16, 1989.*  
*For discounted airfares, call American Airlines at (800) 433-1790 and give them this ID: S-81761.*

*\*To be eligible for a FREE trip to AmiEXPO - Europe, Sept. 15-17, 1989 in Frankfurt, West Germany*  
*Register by Mail - or - Bring This Coupon to the show*  
*or Call 800-32-AMIGA Nationwide (in New York State 212-867-4663)*  
*For Your Ticket to The Amiga Event!*

**Yes, I want to come to AmiEXPO - New York**

☐ Friday ☐ Saturday ☐ Sunday

**Registration is**  
**\$5 Additional**  
**At The Door**

One day - \$15 \_\_\_\_\_

Two days - \$20 \_\_\_\_\_

Three days - \$25 \_\_\_\_\_

**Master Class(es) - List Class and Time - \$50 Each**

\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

Total Amount Enclosed \_\_\_\_\_

NAME \_\_\_\_\_  
COMPANY \_\_\_\_\_  
ADDRESS \_\_\_\_\_

CITY \_\_\_\_\_ STATE \_\_\_\_\_ ZIP \_\_\_\_\_

For ☐ MasterCard or ☐ VISA Payment

Expiration Date \_\_\_\_\_

Account Number \_\_\_\_\_

Name as it appears on card: \_\_\_\_\_

Signature \_\_\_\_\_

Make Check or Money Order Payable to:  
**AmiEXPO 211 E. 43rd St., Suite 301**  
**New York, NY 10017**

**Bring Coupon to the door and get these Pre-Registration Prices!**



Gold Disk's

# James "Max" Morehead

*Max talks about the capabilities, design, and philosophy of MovieSetter.*

**Interview by Rick Rae**  
**Edited by Steve Gillmor**

At the World of Commodore show in Philadelphia, Gold Disk drew a consistent crowd with MovieSetter, their new animation package. James "Max" Morehead, one of the principle programmers on the project, also took the time on several occasions to show the crowd just how easy it is to put together a brief animation using the program. AC had an opportunity to talk with Mr. Morehead about the capabilities, design, and philosophy of MovieSetter.

**AC:** What do you want to tell people about what MovieSetter is and how it differs from the other packages?

**Max:** The emphasis in MovieSetter is on the user interface. It is the simplest of all animation packages on the market,

without a doubt. You don't have to worry about scripts. There is a script inside the program, but it's invisible, completely invisible, to the user.

**AC:** Is it created by the program?

**Max:** It's created by the program; As you are creating your movie, you see what's happening on the screen. For example, you lay down a track of a person walking across the screen, then go back and add sound, add her walking steps. Then when you add another track — say she's walking with her dog — you see her walking with the sound effects happening as you're creating the dog walking. It's not like you decide, "I want this woman to walk," that's one thing, "I want the dog to walk," that's another thing, now you compile them and play it, and "Oh, that's what it looks like!" You see what it looks like all the time. You're always getting feedback. It's WYSIWIG animation.

Another key feature of MovieSetter is the fact that if you have a megabyte you can get minutes of animation, which is realistically impossible without loops on any other animation package. There are two ways to save your movies

in MovieSetter. You can either save all the graphics, all the sound, and all the information the program needs in one file, or you can just save the information the file needs to reload the graphics and sound you are using. For the two-minute demonstration that I showed you, the file that contains all the information the program needs is 40K. The graphics and sound are a few megabytes in this particular example. So when you're saving your movie, it's a matter of a second or two, because it only saves a small amount of information to know what to do with your graphics and sound.

**AC:** So if you can reuse graphics, for example if you have two people sitting and talking, then that's going to be minimal overhead because it's all in the script.

**Max:** Exactly.

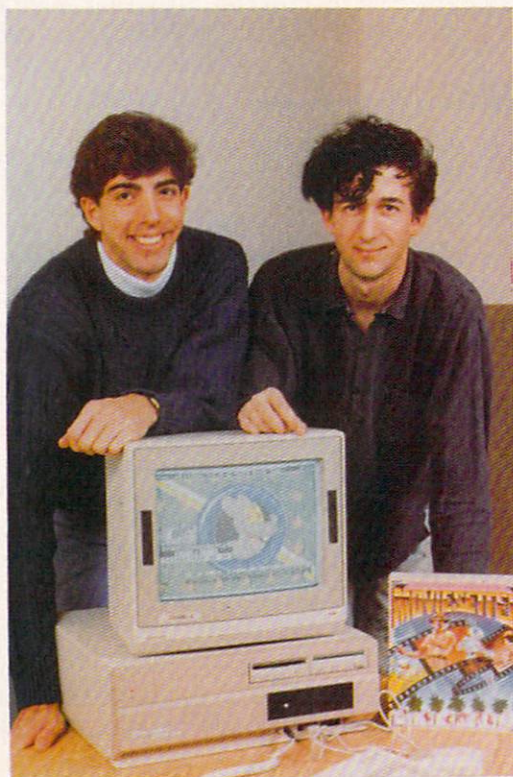
**AC:** The demo that you were doing was really impressive because you were just casually going through the actual creation of a short film. You had a pair of robots walking across the screen; that was what, eight frames in the sequence to make each robot take two complete steps?

**Max:** That's correct. The terminology we use in the program, the robot itself is called a "set"; it's a set of little pictures. Each picture is called a "set element". You group them in logical groups. A dog walking would be a "dog set", and it would have every position he requires to make one full walking step. So with this robot, we had eight elements in that set.

**AC:** Four involved with each foot?

**Max:** That's right, until it repeats.

*Ian Donen  
(left) and  
James "Max"  
Morehead  
(right)*





**AC:** Let's talk a little bit about how the editor is configured because that, to me, was really the impressive thing. Let's assume I am an artist — which is about as far from the truth as we can get — and I'm working on some little production, and I want to do the heroine walking across the screen. I can draw her, and let's say we want to use eight set elements in one complete walking step. Explain to me how I would go into the program and draw those elements, and then how the program causes them to be sequenced.

**Max:** There are two parts to the program: the scene editor, where you create your movies, and the set editor, where you create the images with which you create your movies. So if you're creating the heroine walking, you would go into the set editor and using the drawing tools, which are standard Deluxe Paint-type drawing tools...

**AC:** You said it was almost like a subset of Deluxe Paint.

**Max:** Exactly. We're not pretending to compete with Deluxe Paint on a graphics level, but all these graphics in our demo could have been drawn, and some of them WERE drawn, inside our graphics editor.

So, you go to the set editor, and you want eight positions of her walking. You draw your first one, then you create a new blank element in the set and draw her next position. The program has things like cut, paste, and copy, so if her head is not going to move, you can copy the first one and use that as a template for the remaining seven set elements.

**AC:** Before you left the set editor, you would also do your backgrounds?

**Max:** No. Because the backgrounds can be the size of full video overscan and we don't support that type of editing, you would use another graphics editor. Any one which supports IFF will work. We felt it was best to leave the background editing to software that was designed specifically for that sort of work.

When you're editing your movie, even though you can't see the full background, you can still position your animations anywhere on the full

overscan screen by shifting them off the edge. But when you play your movie you see full overscan, no borders around your picture.

**AC:** I assume you can use digitized images as well? Could you take a digitized image of a lake and have a rowboat travelling across it?

**Max:** Absolutely.

**AC:** So now at this point, you bring up the scene editor, and where do we go from there?

**Max:** Well, since you brought up the lake, let's use that digitized image. We load that in as a background. At this point, that will always be behind everything in the scene. But that brings up an interesting question. Say, for example, we wanted the heroine to walk into the lake. How do I get her to go inside the lake when I can't get her to go BEHIND the lake?

Well, what you would do is, using the scene editor, you would first have her walk across the screen to the point where you want her to enter the lake. You then go into the set editor and bring up your background, because you can still cut pieces of any IFF picture from within the set editor to create new sets. So I cut out a bit of the water, make it a set and call it WATER.MASK; "mask" is a key term in the program.

Now I have a bit of this background as a set, just a little square of it. I go back to the scene editor, and at the point where I want her to enter the lake, I overlay this mask on top of her, so that it appears as though she's entering the water, when in fact...

**AC:** She's actually going BETWEEN the two layers.

**Max:** Exactly, exactly. You can position the layers of sets any way you want to.

As another example, we have an excellent background of a street scene. Say you want someone to enter a building. You'd just cut out enough of the background so that it completely obscures your character and you'd have them walk behind that mask.

**AC:** Now we've pulled in our background and we're ready to have the heroine walk across the screen... into the lake if you'd like. What sequence of steps do you need to go through?

**Max:** Okay, you have your set with the eight set elements. When you use those to create an animation in your movie, you are creating a track, just like in music where the saxophone player plays his part, the piano player plays his part, and the singer sings. These are all separate audio tracks, and then you would mesh them together to create one continuous unit. It's very similar to that concept.

So you would select "new track", select her set, and immediately on your mouse would appear the first image in her set, the first set element. At this point, if you click the mouse, it places her image at the position you clicked and automatically advances to the next set element in the set you are using. You can also have it go backwards, hold, or randomly access any set element, but most likely you'll just go through the order you created them in. And at this point it's just a matter of clicking from the left hand side of the screen to the right hand side of the screen, and she's walked across the screen. At that point the movie is ready to play.

**AC:** As you were doing the robot demo, the program left the previous image in place so you could see how to place the current image.

**Max:** Yes, you can turn on an option called "history", which shows a variable number of previous elements, so you can see the last two elements, the last twenty elements, whatever you need.

**AC:** Now that's one track. You have an unlimited number of tracks?

**Max:** There is no limit whatsoever built into the program. The only limit is 99,000 frames — which we think is a fairly reasonable limit — and the amount of memory. Many programs have problems with chip memory limits. Once MovieSetter runs, there are no extra requirements for chip memory, which means you will never have problems with running out of memory due to low chip memory. It takes the memory it needs when it first runs.

(continued)





## THE HOTTEST AMIGA STORE IN THE COUNTRY.

Amazing Computer Systems is *hot*. Our sizzling selection of Amiga products has become the talk of the town. From more than 1000 software titles in stock to the hottest selections of

hardware, accessories and books. All at red-hot prices. So remember. When you're hot, you're hot. And when you're not, you're not shopping Amazing Computer Systems.

### Amazing Computer Systems, Inc.

Frankford Village Shopping Center  
3030 N. Josey Lane #144 (2 doors South of Skaggs) Carrollton, Texas 75007 (214) 394-8383  
Open Mon.-Sat. 10am-6pm Thurs. 10am-8pm

Amazing Computer Systems is not affiliated in any way with PIM Publications, Inc. or Amazing Computing.

**AC:** Okay, now I decide that I want to give the heroine a little dog, I just do the same thing again?

**Max:** Exactly. You create a set for the dog — you could do it in the same session you did the woman if you knew you wanted it — then just click the dog across the screen. It's WYSIWYG — as you create a track, you see what already exists in the movie. So as you are creating your film, you are in fact playing it in slow-motion, so you always see exactly what it's going to look like. And if you've added sound effects, they work the same way.

**AC:** What is the fastest playback frame rate that people should expect?

**Max:** In Macintosh mode — which means two colors — the program will run between 30 and 60 frames per second. If you have a small ball or something moving around the screen, you'd get 60 frames per second. If you put five or six objects on the screen, it

might slow down to 30 frames per second. So if you want to do pencil sketches in black and white, you can get incredible speed. It won't go much slower than 30 frames per second, no matter how complicated you make it in one bit-plane.

You can control speed in either of two ways. In any frame you can say, "I want the program, if possible, to run at this speed." You always give it a ceiling for the fastest you want it to go. You can't guarantee how fast it's going to go, because you might have sound effects, you might have color cycling, you might have a lot of objects on the screen, all of which can affect the maximum frame rate.

The other way to vary your speed — your visual speed — is the spacing between frames: how far an object moves between frames. If your characters are well drawn — and the characters supplied with the program are designed to run fast in 32 colors — you can still get a smooth, fast appearance in 32 colors, which means

that the movement between each element in your animation is such that it won't look jerky.

The animations that you see on Saturday morning cartoons, many of the new ones run at only four frames per second. It's amazing to me that they get away with it. But if you watch them consciously thinking that it's four frames per second, you can see that that indeed IS how slow they are. So ten frames a second is getting to the point that you can't discern it with your eye anymore.

**AC:** You mentioned the characters supplied with the program; how complete will the selection be? What about the people that aren't artists?

**Max:** The artwork that we supply will allow you to make fantastic movies that don't require you to do any drawing at all.

The one thing about MovieSetter, though, is you always must have the art pre-drawn when you go to create your scene. In other programs you might say, "Make this object small on the left side of the screen, big on the right side of the screen... fill in the blanks." MovieSetter doesn't do that, because we wouldn't be able to get fifteen frames a second if we had to calculate the size increment each frame. But the editor allows you to create a set which goes from small to large, and then when you create your movie it grows as you click on the screen.

**AC:** The audio capabilities I thought were interesting. When you added sound to your robot demo, you put one robot on the left channel and the other on the right channel, just by clicking the gadgets.

**Max:** It's that easy, right. And any IFF sample is compatible with MovieSetter. It goes beyond sound effects, too. If you digitize a piece of music off your stereo, you can play it back in MovieSetter like we did in the demo.

You can say, "Okay, I'll use the one right side channel for my sound effects, I'll use one left side channel all the time for dialog, and the other two channels for the stereo soundtrack." Now if you're talking about a soundtrack, that sort of



sound takes a lot of memory. But with small sound effects, like a ball hitting a bat, a person walking, a gunshot, it takes very little memory. People with one megabyte won't have problems with that.

**AC:** Can you create sets that are larger than the screen?

**Max:** To allow the program to play back smoothly at high speeds, we have to take a fairly large chunk of CHIP memory to do our buffering. In fact, to get its speed, the program triple buffers, which is a bit unusual. To have a set larger than the screen is another 40K of chip we'd have to sacrifice.

We decided we'd like to have sounds running all the time, we'd like to have backgrounds the size of overscan, and we'd like to have unlimited tracks. The best way we found to do that was to put a slight restriction on set size. All that means is that you might have to take a very large thing and split it up into two pieces.

**AC:** Regarding backgrounds... scrolling. You started the robots walking THIS way, then said, "Ah, I don't like that effect, it's too static," and started the background scrolling THAT way, so it looked as if the camera was trying to pan to follow these robots as they moved across the field of view, which was a very nice effect. When you do that, do you take the ends of the scenes and meld them together?

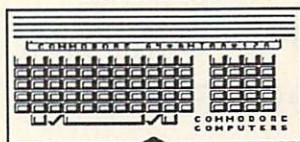
**Max:** That's right. The backgrounds that we will supply will be specifically designed so that when it wraps around, it's an invisible seam. It appears larger than one screen because of that invisible seam.

**AC:** Is it possible to make a really large background if you wanted to pan across, say, a city block and didn't want the same buildings popping up again and again?

**Max:** With this release, no. The constraints were time and memory, and the fact that we'd have to deal with unlimited width. That would require us to copy from fast memory to chip while we were running the animation. Scrolling is a very efficient process right now.

An entire store dedicated to the commodore **AMIGA**.  
Genlocks, Digitizers, Harddrives, Printers, Monitors,  
Modems, Memory Expansion, Books, and Magazines. Along  
with all of the latest software, in stock and on display.  
As always customer satisfaction, and support come first!  
Nothing but the best.

Remember...



The Memory Location

Where

Only **AMIGA** makes it possible.



We're Commodore specialists!  
Only what works!



Open 10-6 Mon.-Thur. 10-8 Fri.  
9-5 Sat. We ship UPS  
Commodore authorized repair in  
store. Low flat rate plus parts.  
Warranty service

**The Memory Location**  
396 Washington St  
Wellesley, MA 02181  
(617) 237 - 6846

But by being tricky with tracks, you can make it appear that your background is wider. For example, say you had mountains in the background. Say you wanted to have a very sparse forest in front of the mountains scrolling along. You could have a number of trees at different depths, and you'd make extra tracks, which make it appear that the background is changing, when in fact the background is the same and the tracks are adding the features.

**AC:** And with the trees, you could run each layer running at different rates, which would give you a tremendous three-dimensional effect.

**Max:** That's right.

**AC:** So the scrolling background is a simple approach that gets you the appearance of movement without having to do any work on your part, and if you want to do something more involved, you use sets.

**Max:** Exactly. The philosophy of MovieSetter is summed up by a favorite quote of mine from Alan Kay: "Simple things should be simple, and complex things should be possible." With MovieSetter, that's definitely the case.

**AC:** Are there any particular things you'd like to highlight about the development of the product?

**Max:** The first thing I want to emphasize about MovieSetter is that it was designed by Gold Disk as a unit. I was given an initial design, I talked about it with everybody at work, and as I worked on the project I got input from everyone. All of Gold Disk deserves credit for this one.

I also want to emphasize that Ian Donen did the graphics editor and set editor; I did the scene editor. He worked for four months on the project. To this point, I have worked five months and I have about two weeks left.

(continued)



# TxE<sup>d</sup> PLUS<sup>tm</sup>

The Text Editor for the Amiga,<sup>tm</sup>  
Plus a whole lot more.

**BlitzDisk** Disk cache, speeds up floppy and  
hard disk reads up to 2000%.

**FastFonts** Speeds up text display.

**FunKeys** Hotkey window manipulator.

**ARP** Latest versions of the AmigaDOS<sup>tm</sup>  
Replacement Programs.

**AREXX** Demo version of the AREXX, the  
macro processor used by TxE<sup>d</sup> Plus,  
that is changing the way people think about computing.



Complete package:  
\$79.95

MC + Visa  
Mass Res. add 5%

**Microsmiths, Inc**

PO Box 561, Cambridge MA 02140

(617) 354-1224 BIX: cheath CIS: 74216,2117

Amiga and AmigaDOS are trademarks of Commodore-Amiga, Inc

From start to finish this project has gone smoothly. We were pretty efficient. The nice thing was that we didn't get halfway through the project and think, "Oh dear, we've made an enormous mistake, we have to change this." The user interface never changed from a point two months into the project.

**AC:** You're obviously pleased with the effort. How do others feel about it? Have you spoken to any professional animators?

**Max:** The animators we've shown it to have said, "Why doesn't everybody do it this way, this is how WE do animation! Why must all computer animators do it in a computer way. Why can't they do it in an intuitive way?"

**AC:** I can see it for an individual who wants to put together just a little animation for fun, or for a user's group demo. But I can also see applications for people who put together instructional disks, presentations for shows, business presentations, and so forth.

**Max:** It would make a fantastic business presentation manager, because if your sets were text that you created in a titling program, you could easily move them around the screen, have transitions to graphs that you've screen-grabbed from another program, and you could integrate a soundtrack easily. Most business presentation managers aren't as easy to use as this. With this program, it's just a matter of clicking on the screen where you want things to go.

**AC:** What about special effects? Can you do fade-ins and fade-outs, for example?

**Max:** Yes. To explain that, I have to talk about "events." A sound is an event because it happens in a specific frame, and ends whenever it ends. In any frame you can have any number of events. A palette change is also an event. So if you wanted to do a fade that you controlled, you could have, say, ten frames where you slowly build a frame from black to full color. If you are intelligent about the way you organize your colors, you could have text you want to fade in in one set of colors, and the rest of your set that

you don't want to fade in in a different set of colors.

**AC:** What other events are there?

**Max:** A background change is an event, because you load a new background and it stays there; scrolling falls in that category; also color cycling. A loop is an event; you can have frames loop any number of times.

You can change the playback rate in every frame. A lot of programs don't let you do that. In our demo, there are points where we decided the characters were just moving too fast. We didn't want to slow it down by hand, so we just changed the playback rate as an event, then changed it back at the end of the scene.

**AC:** What resolutions do you support?

**Max:** The Amiga limits us to low resolution, because in high resolution — 640x400 — there simply isn't enough chip memory to support our playback rate. Playback speed would not be acceptable, and sound would be just about impossible. Remember, though, that low resolution mode is five bit-planes for 32 colors, and it actually goes to 352x240 with full video overscan.

**AC:** What I've seen in your demonstration is that it's more important to have the colors. When you have a static image you can see the jaggies, but when you have something that's moving on the screen, it seems to be less obvious.

**Max:** So far I've only had one or two people comment that it's unfortunate, but they understand that it's still a personal computer, and to get fifteen frames per second in high resolution, the way MovieSetter runs, would be just about impossible without the hardware being faster.

But you know, the fact is that with 32 colors, it's amazing what you can do in low resolution. The people have been impressed by our demo. In terms of programming, switching resolutions is trivial. It's a compromise we had to make for now. There will be a time when the Amiga can support it, and when it's ready, we'll be there

•AC•



# New Products

by Mike Creedon

Other neat stuff

## AmiKit for the Amiga

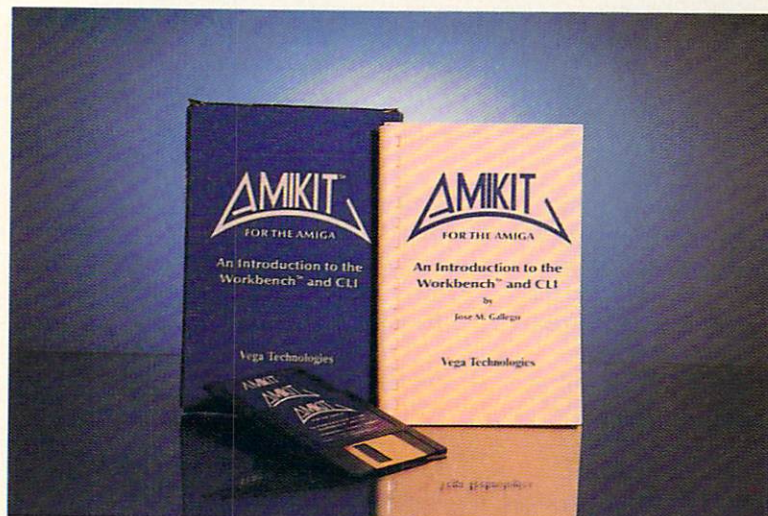
First time computer users who received an Amiga for Christmas will certainly welcome the **AmiKit** as a post-Christmas stocking-stuffer. *Vega Technologies'* AmiKit will help new users get started on the Amiga, as it introduces the Amiga, CLI, and Workbench. The manual's introduction lists a four-fold purpose: to demonstrate some of the Amiga's main features; to introduce new users to the WorkBench; to provide hands-on AmigaDOS training; and to introduce the Amiga Operating System to first time computer users, or those who have upgraded from the C64, C128, or other systems.

The AmiKit is more than just a manual—it includes three disks, one containing the latest version of Workbench, the other two featuring examples from the book, along with some public domain software.

The manual consists of six chapters. In chapter one, new users get an introduction to the Amiga and the AmiKit. Chapter two lists the requirements for using the kit, and also suggests additional equipment and software that beginners may find useful.

Chapter three introduces the WorkBench with its gadgets and icons, explaining how to load the WorkBench, how to format disks, copy files, and customize the WorkBench. In chapter four, the new user begins touring the CLI, with several exercises using CLI commands. (Some exercises are included on the AmiKit CLI disk.) The chapter also shows single drive users how to access other Amiga disks from the CLI mode.

The CLI tour continues in chapter five, with an explanation of some of the



more useful CLI commands. Again, many of the examples are included on the AmiKit disks. To access the wealth of public domain software available on BBS, new users can check out chapter six and its introduction to the modem and public domain terminal programs. Finally, handy appendices list all commands in the AmiKit manual, as well as a description of how to run an AmigaBASIC program, using one of the supplied games as an example.

**Vega Technologies**  
3171 Iris Street  
San Ysidro, CA 92073

## Look It Up

Push over another two beads on the old Abacus—*Abacus Books* that is. Abacus has recently added two more tomes to its growing Amiga library:

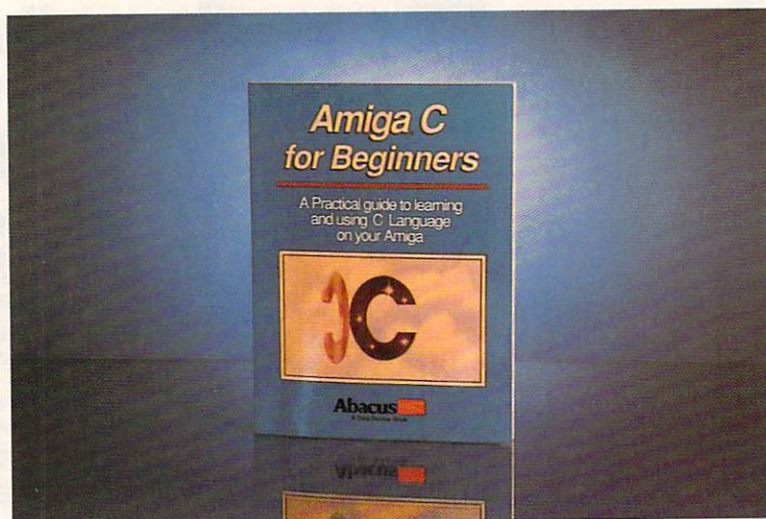
## Amiga C for Beginners and the AmigaDOS Quick Reference Guide.

*Amiga C for Beginners* introduces new Amiga users to the popular C programming language. The book offers a beginner's overview of C, introducing the language and explaining the difference between compiled and interpreted languages. After that, it's time to begin C. Ensuing chapters cover the basics of writing C programs, introducing the editor, compiler, and the linker and demonstrating how the three contribute to the development of C programs. From there, it's on to the first program. Next, take a look at the special features for C; important routines in the C libraries; I/O using C; the scope of the language; tricks and tips; two popular C compilers, and more.

If you're not quite up to C programming on the Amiga yet, maybe you need to take a look at the *AmigaDOS Quick Reference Guide*, a

(continued)





reference guide that will help both beginners and advanced users tap into the Amiga's potential. Three handy indexes will help you quickly find the commands you'll need. Topics covered include all AmigaDOS commands including Workbench, CLI shortcuts, CTRL sequences, ESCape sequences, an Amiga ASCII table, Guru Meditation codes, and error messages. The AmigaDOS Quick Reference is just what you need when you know what you need, but not how to find it.

#### **Abacus Books**

5370 52nd Street SE  
Grand Rapids MI 49058

#### **Papillon, The Later Years**

After an all night programming session, sometimes you feel like you just want to escape. Go for it—but take Bobo with you. Bobo is doing time in that most fearsome of prisons, INEESLAMMER. He's desperate to get out, and you're his only hope. You and his trampoline, that is. Help Bobo and his buddies escape by catching them on the trampoline as they jump out of the window, then try to bounce them over the walls. Do anything - just get them out of INEESLAMMER.

But in order to make sure nothing looks suspicious, Bobo has to attend to the daily drudgery of prison life: dishing out the slop to his prison buddies, washing the floor—which they insist on

walking over, and mashing up 'taters for the evening swill—Uh, sorry, Mr. Guard, I mean the evening meal. For all his tasks, Bobo will need your help to keep his cover.

You'll also have to help him cover his tracks—literally. Bobo's warden is a wily old sentry, and if Bobo leaves just one footprint behind, he'll never escape.

Bobo comes from the French company Infogrames, and little Bobo has a bit of French mystique about him—by way of Inspector Clouseau. Bobo's stumbling stumblebum—Or is it stumbling bumblebum? Bumbling stumbler?—antics take you back to the glory days of Clouseau. The sound effects are great. The comic boings and the smashing, crashing glass will make you want to send one of Bobo's little buddies for a fall just so you can hear what happens. But be nice to little Bobo. Someday, if you're not careful, you just might end up INEESLAMMER.

#### **Infogrames**

84 rue du 1er Mars 1943  
69628  
Villeurbanne CEDEX-France

#### **The Talking Animator**

JMH Software has something for all you creative souls out there. In fact, The Talking Animator can even add a little creativity to lower forms of creative life. (Like Amazing copy editors, for example.) The Talking Animator is a

presentation tool anyone can use, for anything from a handwritten note to a full-blown graphics presentation. The program begins with an easy-to-use word processor to get you started on presenting your idea. But there's more.

A palette of colors on the right side of the screen will help you add some color to your presentations. All the colors can be adjusted to any one of 4096 colors. The right mouse button controls a large paint brush, while the left button gives you a fine brush or a pen for those moments of creative calligraphy. Press the shift key along with the mouse, and you'll get a fill-in of the area selected. You can also mix text on the same page as mouse-drawn pictures. And clicking on the TALK icon will give you a spoken version of the text as it's written on the screen.

To create page-flip animation, just use the COPY icon to copy the current page's graphics to the required number of pages. Image ghosting helps ensure that you place your graphics just right. When you're ready to roll—uh, flip—just click on the FLIP icon to get things moving. The computer will stop on each page just long enough for the computer to read the on screen text. The program includes the usual LOAD, SAVE, PRINT, and CLEAR icons, plus a HELP icon, which will speak out the function of the icon the mouse is pointed to. So you don't have to read the documentation, just listen.

#### **JMH Software of Minnesota Inc.**

7200 Hemlock Lane  
Maple Grove MN 55369

#### **Well, my Broker is...**

Budding stock market tycoons, *Free Spirit Software* has something for you. **The Securities Analyst** helps individual investors chart any number of stocks for as long as they would like. The analyst helps investors make the right buy or sell decision at precisely the right moment, based on scientific analysis of market trends.

The Analyst offers several forms of analysis: Moving Average, Accumulation/Distribution, Relative Strength, Performance, Point and Figure, Trailing Stops, Stock Chart, Momentum, and Price Analysis. It also allows graphs to be displayed on the screen or dumped into



a printer. The manual includes a technical analysis of stock market trends, featuring an explanation of the three schools of thought on stock selection, an explanation of the different analysis methods, and a chart to help you decide which method is best for your type of investment.

Once you get the system pegged down, give me a call.

**Free Spirit Software, Inc.**  
P.O.Box 128/58 Noble Street  
Kutztown PA 19530

### **Creativity Deluxe**

If you're like me, when you want to send a "personal" greeting card, you go the local card shop and buy one. If you need to announce something like a yard sale or garage sale, a hand-scrawled sign will usually be the best you can do. Business cards are work for a professional—you could never do a decent job with something like that. Well, it's time to change your way of thinking. **DeluxePrint II** from the *Electronic Arts's* Deluxe Creativity Series lets you add a personal touch to all those little print jobs that you previously might have either sent out to professionals or made do with a slightly sub-artistic effort on your part. **DeluxePrint II** lets you print greeting cards, calendars, flyers, bumper stickers (One shudders at the possibilities here.), posters, menus, banners, labels, etc. Just about anything you need to print, **DeluxePrint II** can help you print.

The package includes a program disk, manual, and a free bonus art disk with 30 extra large graphic images, 20 **DeluxePaint II** images, and 30 complete, pre-built cards, banners, and calendars.

**DeluxePrint II** features 7 ready-made page formats for signs, banners, cards, labels, letterhead, and more. It allows you to use graphic images and borders with up to 32 colors from a palette of 4096. Add flair to your print jobs by using images from **Deluxe Paint II** or other graphics programs, or create and edit your own multi-color graphics with the built-in multi-color graphics and border editor with **DeluxePaint II** style tools, shapes, and fills.



You can mix several different font styles, sizes, or graphics in one printout, flip borders and graphics 4 ways for special effects, or print in either color or black and white.

**Electronic Arts**  
1820 Gateway Drive  
San Mateo CA 94404

### **Three Digit fun**

It's been said that Michelangelo could tell by looking at a piece of granite what sculpture lay beneath it; his job was just to chip away the excess stone to reveal the image. Well, **Powerstyx** takes that idea and gives it a little twist—well, maybe it's more like a wrench.

In **Powerstyx**, you aren't chipping away at granite to reveal a masterpiece, you're battling through nasty gnawing skulls, slicing scissors, and cruel crosses as you battle the clock. Your object: to conquer these obstacles and reveal the masterpiece hidden beneath each horrific screen. **Powerstyx** features 15 different levels, each containing 128 colors, with fast graphics and full use of the Amiga's sound. So if you're feeling artistic—and brave—give **Powerstyx** a try.

If you're still in the mood for an amazing time after fighting through the obstacles of **Powerstyx**, why not embark on the final mission? In **Final Mission** you must fight through the Labyrinth of terror, the last stage of insanity. (So *that's* what they call it.) To get through the Labyrinth, you must avoid the mines, traps, and energy fields while reaching

toward the red and yellow balls of madness and insanity that will bring you closer to freedom. If all else fails, shoot your way out. **Final Mission** features 20 levels of terror on the last stage of insanity. Enter if you dare.

After you chip away at a macabre masterpiece and fly through the Labyrinth of terror, you'll want something to soothe your frayed nerves. **Hole-in-One Miniature Golf** is just what the (systems) analyst ordered. The game features 72 holes on 4 separate courses, and it offers traditional mini-golf along with some hilarious surprises. (Pinball golf?) Sound effects like the roaring crowd, the sailing ball, and your bad shots will add to the fun—or aggravation.

**Hole-in-One** brings you right down to the green—or the traps. Different elevations are noted as light color for higher elevations and dark for lower elevations. The contour menu lets you view the course from different perspectives. Replay lets you review your last shot; with retry, you can try the shot again. The game supports up to 4 players, so you can battle it out with a few pros. Drive 'em wild.

All three games distributed by:

**DigiTek, Inc.**  
10415 N. Fla. Ave. Suite 410  
Tampa, FL 33612

•AC•



# TORCH

*reviewed by Jeffery Scott Hall*

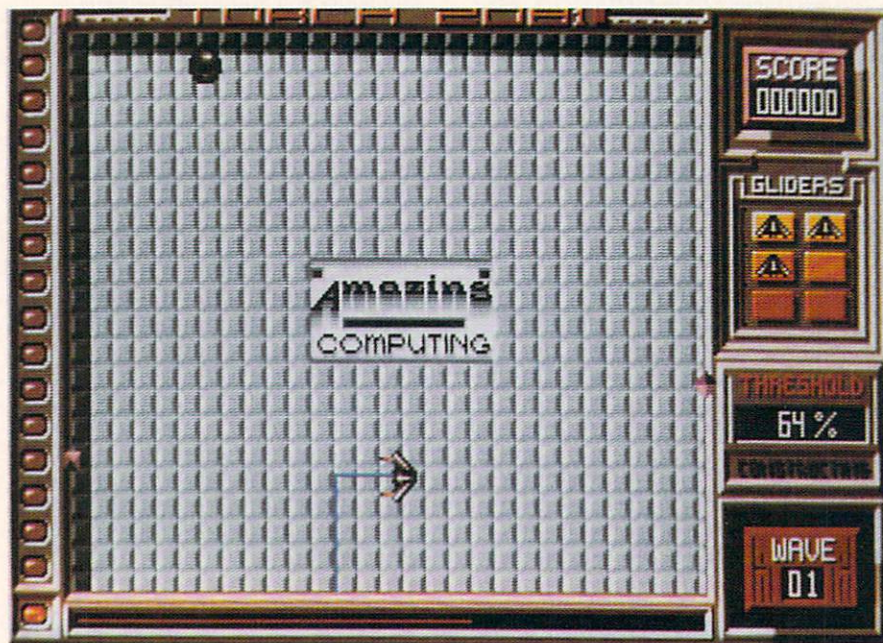
Torch 2081, from Digital Concepts, is a reincarnation of the arcade game Qix, with many enhancements designed to take advantage of the Amiga. Once the game loads, you will see a title sequence, at which time you may either abort by pressing any key or wait for the sequence to finish. Once this is over, you will be taken to the skill level menu, from which you can scroll through levels ranging from novice to crazy. This allows beginning players to start at level one and work their way up, while giving seasoned players the chance to skip easier levels and advance to more difficult rounds. Now, let's explore Torch 2081 for those whom have never played Qix.

### **What is TORCH 2081?**

The game takes place in the year 2081 when interplanetary nuclear war has all but destroyed our planet Earth. The whole solar system is falling apart before our very eyes. The planets are falling from space onto our lovely Earth—which has been given the nickname Torch because of the massive fireballs of destruction. Trying to survive this horrible ordeal, you have been sent out as our last hope. Your goal is to construct as many metallic shields as possible some 10 miles above the planets surface. With these shields in place our remaining population will survive.

The object of the game is to maneuver your ship around the border of the playfield. You will be provided with three ships to start with, and a bonus ship at every 10,000 points. To move the ship to different areas, you must hold the fire button while pressing the joystick in the direction you wish to go. A construction line will trail behind you, containing the energy you need to create the metallic shields. Once this trail has been completed by making contact with several points on the border, this will then become energized creating a metallic shield. After the first one is created, you have a choice of continuing to build shields by making contact with the borders or you can build onto the shields already created.

Whether you fly around this area or sit still, your ship will use up a certain amount of fuel. If you run out of fuel, your ship will be destroyed. This game contains an amazing 99 levels of play, with each one progressively tougher than the last. Each level has a completely different screen and color in which you construct the shields. You are required to





# 2081

shield a certain percentage of the area you're in, which varies according to the level.

To make constructing your metallic shields more difficult, you must avoid objects which will destroy the ship. There are three basic kinds of obstacles: fireball, spitfire, and perimeter patrollers. The fireball is the player's main obstacle in the game, and it will elude, change speeds, travel in different directions, and has no set pattern. Spitfires are offsprings from the fireball's core. These smaller balls of fire will travel in a set, straight pattern homing in on your ship. While the last obstacles, the perimeter patrollers, only exist on the screen's border, they are still fatal if they make contact with your ship.

## **Bonus levels and awards**

After you complete 5 levels in the game, you'll be given a bonus level. Every bonus level has two fireballs, which you must either separate or capture by constructing shields. If you separate the fireballs you will be rewarded with ample points; capturing gives you a bonus ship as well as the extra points. Points are also given for the percentage of shields constructed on the screen.

Awards appear at certain times in two colors: yellow and orange. The yellow awards offer many special features: extra ship, activate ship's shield so you can't die, warp to next bonus level, give three extra fuel units, decreases area by 5% allowing for a lesser percentage you must cover, warp to a higher level, and bonus points from 500-2500. The orange "awards" represent

hinderances, not helps. These include ship destroyed, lose 3 units of fuel, add 5% to area, and lose points.

## **Status panel**

To the far right of the screen is the status panel. On it you will find your current score, a graphic representation of the number of ships remaining, and the percentage of the area you must complete. The far left side of the screen contains your ship's fuel indicator displaying how much is currently available. A red line at the bottom of the screen indicates how much of the area is left to complete, while a yellow line above it shows the amount already completed.

## **Summary**

Torch 2081 is one of the best strategy-action games I have ever seen for the Amiga. To help during game play a handy pause option is included, which allows you to take a break and resume play later. Also, a top ten function saves the high scores to disk (provided the disk is write enabled). The manual gives you complete instructions and a brief story line.

Game play is fast and difficult—I only made it to level 11 after some help from the company. I wish to thank Digital Concepts for giving so much thought to the game. An arcade product of this kind really shines because of the expert programming they've done. Game play is enjoyable, combining great graphics and sound effects. If you have played Qix in the arcades or if you're looking for an enjoyable game which will keep you coming back for more, then Torch 2081 is definitely for you.

## **Hints and tips**

*The basic strategy of Torch 2081 is to build small boxes on top of each other until you reach either the top or side of the screen. This will allow you to fill in more of the screen area with a lesser chance of getting hit.*

Always avoid orange "awards" while playing. For higher score, go for the orange T when it appears. This will increase the percentage of the area you must fill in, giving you more points.

On the bonus rounds, try to complete as much of the area as possible. Then separate the two fireballs by drawing a line between them and connecting it to one of the sides, creating a metallic shield, and giving you many bonus points.

**Torch 2081** \$29.95

## **Digital Concepts**

28800 Gratiot Ave. Suite 162  
Roseville, MI 48066  
(313)-772-1416



# Send us your best...



The pages of Amazing Computing™ are filled with the efforts of Amiga enthusiasts just like you. Each article in AC is the written record of a fellow Amiga user who has learned a little more about this amazing computer. The exception is, the fellow who wrote the article was paid for his efforts.

Amazing Computing was created for the Amiga community to share and expand the knowledge of the Amiga. From Hardware "how to articles" to Software reviews, each Amazing Author has taken our knowledge a little farther. Each article has given us a little more insight into how our Amigas can make us more creative and more productive.

If you have discovered a new use for your Amiga or you have created a great program, let us know. Not only are your ideas important, they are necessary. Amiga users enjoy reading about other users creating a better life with their Amigas.

AC is continually searching for all articles on the Amiga. We are searching for technical articles and everyday solutions. We enjoy game reviews and your insights into the latest compilers. In short, if it is Amiga related or of interest to the Amiga community, we want to publish it.

For a free copy of Amazing Computing's writer's guide, please write to:

Amazing Computing Writer's Guide  
PiM Publications Inc.  
P.O. Box 869  
Fall River, MA 02722

Or call 1-508-678-4200 and ask for a copy.

There are no ideas too small or too large to be shared and you will never know how great it feels to share your ideas, until you have been published.



# Roomers

by the Bandito

*[The statements and projections presented in "Roomers" are rumors in the purest sense. The bits of information are gathered by a third party source from whispers inside the industry. At press time, they remain unconfirmed and are printed for entertainment value only. Accordingly, the staff and associates of Amazing Computing™ cannot be held responsible for the reports made in this column.]*

Word from Westchester has reached the Bandito that Joel Shusterman, Commodore's vice president of marketing, has turned in his resignation to Max Toy. Shusterman has only been at Commodore a short time. Formerly, he was the founder and president of Franklin Computers, maker of Apple II clones. No reason was given for the resignation, but the Bandito wonders if it could have been impatience with the Commodore bureaucracy. Developers spoke highly of Shusterman as a straight shooter who understood their problems, unlike some of the other people at Commodore.

*[Commodore confirmed that Mr. Shusterman resigned. Concerning the rest of the Commodore material in this month's Roomer's, Commodore said "they don't respond to rumours."—Ed.]*

The Bandito also received data about Commodore's latest managerial move. It seems they've cut their district sales managers salaries by \$1200 a month, which has the DSM's howling. Commodore says that they can make up the difference through commissions, with the intent being that the DSM's get out and hustle their butts to prove how worthwhile they are. Sounds like a good bottom-line strategy (pun intended), but the timing and the presentation leave

something to be desired. If you listen close, you can hear the sound of résumés being polished...

On another front, some developers are sore that Commodore doesn't support desktop video from third-party vendors. They point out that Apple is pushing desktop video on the Mac II very hard, giving out lots of marketing dollars, PR help, and so on to get more desktop video hardware and software on the Mac II. Meanwhile, Commodore ignores the numerous high-quality genlocks like SuperGen, the professional video titling software like Pro Video CG1, and other video products already out on the Amiga that make it a true video machine. Commodore is busy hyping their Professional Video Adapter (PVA), which takes up three boards in an A2000 to give you frame-grabbing, genlock, and maybe some editing software. The PVA boards won't be out till the end of the year, and those who've seen it wonder why Commodore bothers when the Video Toaster does all that and more for a lower price.

It seems like Commodore is suffering from NIH (Not Invented Here) syndrome, and not pushing the products from third-party vendors that make the Amiga unique. The Bandito hopes Commodore realizes desktop video is where the Amiga can really shine against other computers, and that they stop trying to compete against desktop publishing and paint programs. Go for where the enemy is weakest, not where they're strongest.

## World of Commodore Philadelphia Report

Saturday the show was packed, but the other days were dead, according to exhibitors. But at least it was busier than LA AmiEXPO, which surprised many exhibitors who have attended the string of successful AmiEXPOs. For the eager

customer, there were no terribly hot new products, but it was a good way to see at one time everything on the market, and to listen to PR people tell you about their new product that's going to ship at any microsecond. (Or is that megasecond? PR types always do have trouble with those nasty little prefixes.) Besides, the entry fee was lower than AmiEXPO.

Why are fees getting so high for trade shows? The trade show business is getting bigger, and therefore bigger profits are expected. The promoters aren't in it for their health, after all. But maybe they could charge the companies a little more to exhibit or even take a percentage of their sales to help reduce the cost to the consumer.

## World of Commodore Toronto

This show, the original version of the Philadelphia event, drew over 50,000 people, making it the biggest computer show ever in Canada. No-shows included Aegis, Mimetics, and MicroIllusions. For hot products: Dragon's Lair finally arrived—initial reports are that the game has some great scenes, but disk swapping is a pain (hard disks are wonderful, aren't they.) It's one of the hottest selling games for this Christmas. Hot parties: Gold Disk celebrated a great year and the launch of MovieSetter—the name sounds like a famous canine actor, "Rex, the Movie Setter." MovieSetter was a big hit at the show; it's an animation package (which makes more than a dozen by the Bandito's count) with a number of interesting features. A demo disk and some full-page advertising are now making the round—the software itself should be available as you read this.

## Hot demos

DeluxePaint III was a star attraction, as was the new Video Toaster demo tape. NewTek premiered Digi-

(continued)



## Even Up The Score!

### INVESTOR'S ADVANTAGE

Let your Amiga give you the Advantage in making *better investment decisions!*

Color graphics of Individual Stocks and General Market Trends help you make more profit in this volatile market. High Low Close, Moving Averages, Centered Moving Averages, Volume, Relative Strength, Stochastics, Wilder's RSI, Cycles, Trend lines and Momentum. Powerful reports such as the Relative Strength Report help you pick the best performers. Use the Market Barometers to help you time your market entries. Update Stocks, Mutual Funds and Commodities manually or automatically. Easy to use communications included.

Only **\$99.95**

See your local Dealer or Call:

Software Advantage Consulting Corporation  
37346 Charter Oaks Blvd  
Mt. Clemens, MI 48043 (313) 463-4995

Amiga and the Investor's Advantage are trademarks of their respective companies.

View Gold — a revised version of the Digi-View hardware and software designed specifically for the A500 and A2000 (no more gender bender needed). Other bits of news collected at the NewTek booth: The widget for letting Digi-View take color pictures directly from a VCR, originally code-named Digi-Adapte—sounds like something that lets you plug your finger into a socket—has turned into a much more elaborate device, with more features than just getting color pictures from a VCR. But it won't be out until after the Toaster debuts, so don't hold your breath waiting for it. However, Digi-Paint II is getting closer to shipping, and it should appear in the first quarter of the year to revive the now-dormant HAM Paint Wars.

*(According to NewTek, they have been developing Digi-Paint II for over 18 months, and it will be shipping now (sic) —Ed)*

The big trade show question for the publishers: What shows do we support? Many regional and European shows are popping up, and the number

of shows increases every month.

Attending a show takes a lot of time and effort, especially for a small company (which most Amiga publishers are). The Bandito hears that all the publishers will not attend all shows, so the competition for exhibitors and attendees will get fierce this year.

#### At Comdex

The show is getting monstrous, it's held not only in the convention center but at practically every hotel in town. The Commodore booth seemed like the lone Amiga outpost amidst an army of IBM products. There's talk of splitting off Macintosh products into a separate Comdex, leaving the original Comdex just for IBMs and the occasional outsider like Amiga.

But despite the show's overwhelming IBM emphasis, the Commodore booth got good traffic. Octree Software showed off a Caligari pro version for beaucoup bucks; apparently, the existence of this pro version is part of the reason that their EA product has yet to appear. The story appears to be long and involved, but

there's still hope that we can see a version of Caligari at a price everybody can afford. According to those who've seen it, the program's interface is superb. MicroIllusions showed off Photon Cel Animator, but their MusicX package is still not shipping. This is good news for the other publishers of sequencer software, because the beta-testers of MusicX say it should easily beat all the other software on the market.

But the most interesting stuff was from Commodore: They showed a transputer prototype, an A500 hard disk and expansion chassis, and a 1024 x 800, 256 colors out of 16.7 million palette graphics card. On the personnel side, the word is that there's still some "dead wood" in Commodore that's holding them back, but that should be pruned by the springtime.

Will the Amiga 3000 be appearing in 1989? Pressure is building for Commodore to come out with a 68020 machine, especially since the competition is moving to the 68030 already. But the marketplace is looking for more significant changes than just a 68020. The Amiga 3000, Commodore's response to this demand, won't appear until late in the year, at the earliest. Look for more colors, higher resolution, a built-in SCSI port, and a socket for a math co-processor (or maybe one provided). Higher-density disk drives are also in the works. But Commodore is in no hurry to get it to the market as long as A2000 sales are good, figuring that that the 3000 will just take their time. Seems to the Bandito that Commodore should try to up the technological ante, rather than waiting to respond to everyone else.

#### Hit the road, Cratchett.

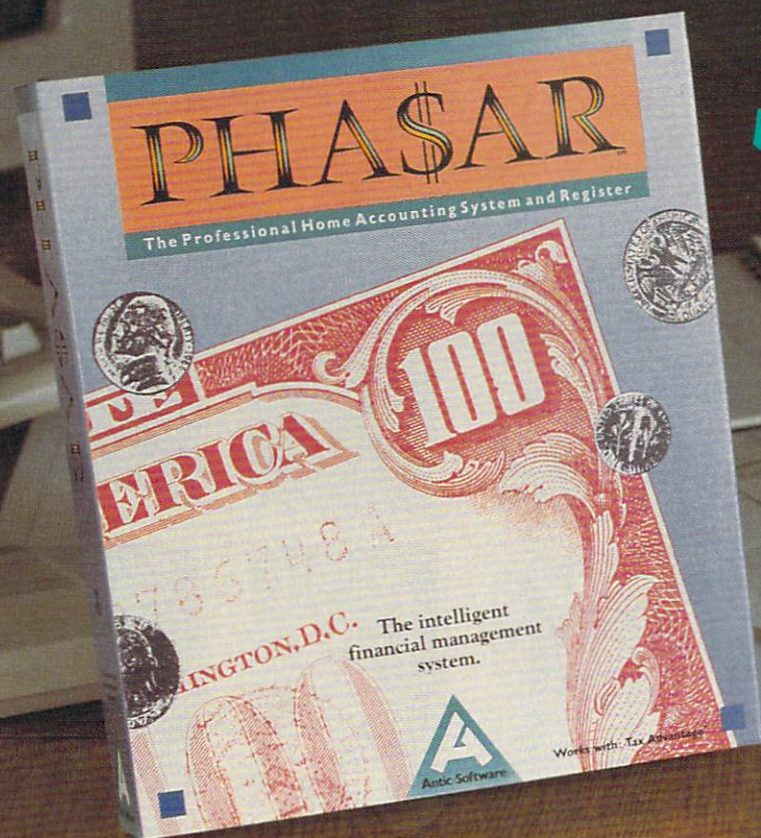
The Christmas season, traditional time of peace and goodwill toward men, isn't so fun for some of the people working at big software companies. In the grand Christmas spirit of Ebenezer Scrooge, Electronic Arts, (see ed. note) and Mediagenic both announced layoffs (or "restructurings", or just "personnel changes"). The winner was Mediagenic, with a total of 30 employees hitting the road.

*(The Bandito's original story mentioned two other companies; however, when we checked with those companies, we found that not only were there no layoffs, the companies had*

*(continued on page 59)*



# The Sensible Solution



New!  
Version 3.0

1988 Amiga World  
Reader's Choice Award

## PHASAR

**Don't get lost in your finances—now you've found the intelligent financial management system.**

The best personal accounting software package has just gotten better.

Introducing PHASAR version 3.0. Updated and improved to take full advantage of your PC's friendly user interface, PHASAR will take away the drudgery of error-prone manual accounting.

By incorporating more powerful analysis, reporting, and financial functions, PHASAR makes it easier than ever to manage your money like a professional.

**PHASAR 3.0...** the only way to make sense of your personal finances.

Antic Software and Antic are trademarks of Antic Publishing, Inc.  
Amiga is a registered trademark of Commodore-Amiga, Inc. \*Atari ST is a registered trademark of Atari Corp. © Copyright 1987 Marksman Technology, Inc.

### New Features:

- Enhanced online calculator with 5 level memory buffer
- Sorting by account name or number
- Improved tax program now doubles as report writer
- Custom define screen colors
- Recall most recent transactions
- And many more...

### General Features

- 130 expense/income categories
- 40 accounts (VISA, checking etc.)
- Up to 500 transaction per month
- Artificial Intelligence parser predicts input, minimizes typing
- Instant context-sensitive help function
- Relational design integrates financial data

### Transactions

- Create macros for repetitive entries
- Point-and-type spreadsheet-style relational summary table

### Reports and Analysis

- Analyze loan/savings plans
- Clearly formatted printed reports include: Account summaries, Category summaries, Net worth statement, Tax calculations
- Sort categories in any order
- Built-in data graphing

### Tax

- Completely integrated tax preparer and report generator

### Date/Address Organizer

- Online organizer stores up to 250 names, addresses and phone numbers
- Integrated scheduler stores special occasions.
- Event alarm feature

**Amiga® or Atari ST® version: \$89<sup>95</sup>**

Available at your favorite store or order Toll Free by phone:  
(800) 234-7001

Antic Software, 544 2nd St.,  
San Francisco, CA 94107  
(415) 957-0886

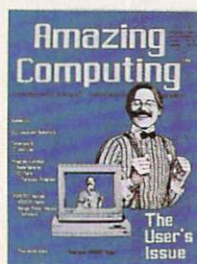


MM8AW-4



# Amazing COMPUTING™

## • EXPANDING REFERENCE •



VOLUME 1.1



VOLUME 1.2



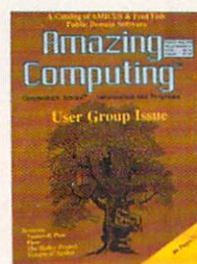
VOLUME 1.3



VOLUME 1.4



VOLUME 1.5



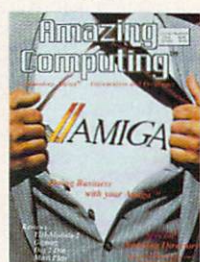
VOLUME 1.6



VOLUME 1.7



VOLUME 1.8



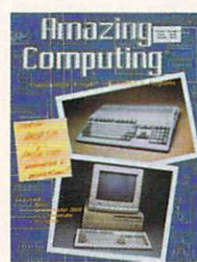
VOLUME 1.9



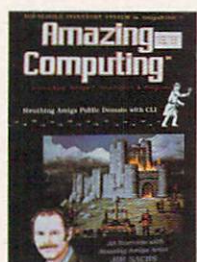
VOLUME 2.1



VOLUME 2.2



VOLUME 2.3



VOLUME 2.4



VOLUME 2.5



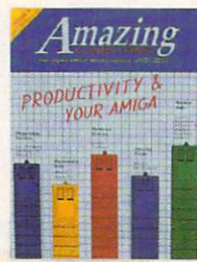
VOLUME 2.6



VOLUME 2.7



VOLUME 2.8



VOLUME 2.9



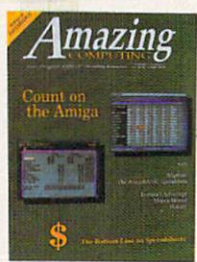
VOLUME 2.10



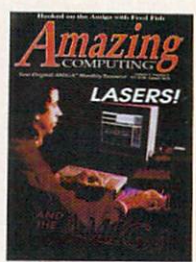
VOLUME 2.11



VOLUME 2.12



VOLUME 3.1



VOLUME 3.2



VOLUME 3.3



VOLUME 3.4



VOLUME 3.5



VOLUME 3.6



VOLUME 3.7



VOLUME 3.8



VOLUME 3.9



# Your Resource to the Commodore Amiga™

Expanding reference is not just an empty promise. The pages of *Amazing Computing™* are filled with articles on technical operations and procedures, basic use, and just-plain-fun. The growing library of *Amazing Computing's* Back Issues contains articles ranging from building your own IBM Disk controller, to setting up your own startup sequence. *Amazing Computing™* has repeatedly been the first magazine to offer the Amiga users solid, in depth reviews and hands on articles for their machines.

## From the Beginning

Since February 1986, *Amazing Computing™* has been providing users with complete information for their Amigas. This store house of programs and information is still available through our back issues. From the Premiere issue to the present, there are insights into the Amiga any user will find useful. AC was the first magazine to document CLI, tell its readers how to connect a 5 1/4 IBM drive, describe a 1 meg upgrade hardware project for the A1000, and many more. Please read the list of topics AC has covered below to find the information you have been missing.

**Back Issues are \$5.00 US, \$6.00 Canada and Mexico, \$7.00 Foreign Surface**  
All payments must be made by check or money order in U.S. funds drawn on a U.S. Bank.

## Limited Supply

Unfortunately, nothing lasts forever, and the availability of some of our Back Issues is definitely limited. Complete your *Amazing Computing™* library today, while these issues are still available, by completing the order form in the back of this issue.

### Volume 1 Number 1 Premiere 1986

**Super Spheres** By Kelly Kaufman An ABasic Graphics prog.  
**Date Virus** By J. Foust A disease may attack your Amiga!  
**EZ-Term** By Kelly Kaufman An ABasic Terminal program  
**Mega Mania** by P. Kivowitz Programming files & mouse care  
**Inside CLI** by G. Musser a guided insight into the AmigaDOS™  
**CLI Summary** by G. Musser Jr. A list of CLI commands  
**AmigaForum** by B. Lubkin Visit Compuserve's Amiga SIG  
**Commodore Amiga Development Program** by D. Hicks  
**Amiga Products** A listing of present and expected products

### Volume 1 Number 2 March 1986

**Electronic Arts Comes Through** A review of software from EA  
**Inside CLI** part two G. Musser Investigates CLI & ED  
**A Summary of ED Commands**  
**Live!** by Rich Miner A review of the Beta version of Live!  
**Online and the CTS Fabfile 2424 ADH Modem** by J. Foust  
**Superterm V 1.0** By K. Kaufman A term. prog. in Amiga Basic  
**A Workbench "More" Program** by Rick Wirth  
**Amiga BBS numbers**

### Volume 1 Number 3 April 1986

**Analyst** a review by Ernest Vivaldo  
**Reviews of Raster, Baratas and Mindshadow**  
**Fort!** The first of our on-going tutorial  
**Deluxe Draw!** by R. Wirth An Amiga Basic art program  
**Amiga Basic: A beginners tutorial**  
**Inside CLI** part 3 by George Musser George gives us PIPE

### Volume 1 Number 4 May 1986

**SkyFox and Artisoft Reviewed**  
**Build your own 5 1/4 Drive Connector** By Ernest Vivaldo  
**Amiga Basic Tips** by Rich Wirth  
**Scrimper Part One** by P. Kivowitz prog to print Amiga screen  
**Microsoft CD ROM Conference** by Jim O'Keane  
**Amiga BBS Numbers**

### Volume 1 Number 5 1986

**The HSI to RGB Conversion Tool**  
by S. Pietrowicz Color manipulation in BASIC  
**AmigaNotes** by Rick Rae The first of the Amiga music columns  
**John Foust Talks** with John Foust A first "under the hood"  
**Sidecar** with R. J. Mical at COMDEX™  
**How does Sidecar affect the Transformer**  
an interview with Douglas Wyman of Simile  
**The Commodore Layoffs** by J. Foust A look Commodore "cuts"  
**Scrimper Part Two** by Perry Kivowitz  
**Marsauder** reviewed by Rick Wirth  
**Building Tools** by Daniel Kary

### Volume 1 Number 6 1986

**Temple of Aphrodite** reviewed by Stephen Pietrowicz  
**The Halley Project: A Mission** reviewed by S. Pietrowicz  
**Flow** reviewed by Erv Bobo  
**Textcraft Plus a First Look** by Joe Lowery  
**How to start your own Amiga User Group** by William Simpson  
**Amiga User Groups**  
**Mailing List** by Kelly Kaufman a basic mail list program  
**Pointer Image Editor** by Stephen Pietrowicz  
**Scrimper** part three by Perry Kivowitz  
**Fun With the Amiga Disk Controller** by Thom Sterling  
**Optimize Your Amiga Basic Programs for Speed** by Pietrowicz

### Volume 1 Number 7 1986

**Aegis Draw: CAD comes to the Amiga** by Kelly Adams  
**Try 3D** by Jim Meadows an introduction to 3D graphics  
**Aegis Images/Animator** a review by Erv Bobo  
**Deluxe Video Construction Set** reviewed by Joel Lowery  
**Window requesters in Amiga Basic** by Steve Michel  
**ROT** by Colin French a 3D graphics editor  
**"I C What I Think"** Ron Peterson with a few C graphic progs  
**Your Menu Sir!** by B. Catley program Amiga Basic menus  
**IFF Brush to AmigaBasic "BOB"** Basic editor by M. Swinger  
**Linking C Programs with Assembler Routines** by Gerald Hull

### Volume 1 Number 8 1986

**The University Amiga** By G. Gamble Amiga at Washington State  
**MicroEd** a look at a one man army for the Amiga  
**MicroEd, The Lewis and Clark Expedition** reviewed Fritzele  
**Scritble Version 2.0** a review  
**Computers in the Classroom** by Robert Fritzele  
**Two for Study by Fritzele Discovery & The Talking Coloring Book**  
**True Basic** reviewed by Brad Grier  
**Using Fonts with the Amiga**  
**Marble Madness** reviewed by Stephen Pietrowicz  
**Using Fonts from AmigaBasic** by Tim Jones  
**Screen SaVer** by P. Kivowitz A monitor protection prog. in C  
**Latitude MAKE Utility** reviewed by Scott P. Evenden  
**A Tale of Three EMACS** by Steve Poling  
**Jmap File Reader in Amiga Basic** by T. Jones

### Volume 1 Number 9 1986

**Instant Music** Reviewed by Steve Pietrowicz  
**Mindwalker** Reviewed by Richard Knipper  
**The Algebra Memory Board** Reviewed by Rich Wirth  
**TeXEd** Reviewed by Jan and Cliff Kent  
**Amazing Directory** A guide to the sources and resources  
**Amiga Developers** A listing of Suppliers and Developers  
**Public Domain Catalog** A listing of Amigos and Fred Fish PDS  
**Dos 2 Dos** review R. Knipper Transfer files from PCMS-DOS  
**MaxPlan** review by Richard Knipper The Amiga Spreadsheet  
**Gizmoz** by reviewed by Peter Wayne Amiga extras!  
**The Loan Information Program** by Brian Catley  
basic prog. to for your financial options  
**Starting Your Own Amiga Related Business** by W. Simpson  
**Keep Track of Your Business Usage for Taxes** by J. Kummer  
**The Absolt Amiga Fortran Compiler** reviewed by R. A. Reale  
**Using Fonts from AmigaBasic, Part Two** by Tim Jones  
**68000 Macros on the Amiga** by G. Hull Advance your ability.  
**TDI Modula-2 Amiga Compiler** review by S. Fawciszewski

### Volume 2 Number 1 1987

**What Digi-View is... Or, What Genlock Should Bel** by J. Foust  
**AmigaBasic Default Colors** by Brian Catley  
**AmigaBasic Titles** by Bryan Catley  
**A Public Domain Modula-2 System** reviewed by Warren Block  
**One Drive Compile** by Douglas Lovell Latios C with one drive  
**A Megabyte Without Megabucks** by Chris Irving  
An Internal Megabyte upgrade  
**Digi-View** reviewed by Ed Jakob  
**Defender of the Crown** reviewed by Keith Conforti  
**Leader Board** reviewed by Chuck Raudons  
**Roundhill Computer System's PANEL** reviewed by Ray Lance  
**Digi-Print**...by New Tek reviewed by John Foust  
**Deluxe Paint II**...from Electronic Arts reviewed by J. Foust

### Volume 2 Number 2 1987

**The Modem** by Joseph L. Rothman efforts of a BBS Sysop  
**MacroModem** reviewed by Stephen R. Pietrowicz  
**GEMINI or "It takes two to Tango"** by Jim Meadows  
Gaming between machines  
**BBS-PCI** reviewed by Stephen R. Pietrowicz  
**The Trouble with Xmodem** by Joseph L. Rothman  
**A Disk Librarian in AmigaBASIC** by John Kennan  
**The ACO Project...Graphic Teleconferencing on the Amiga**  
by S. R. Pietrowicz  
**Flight Simulator II...A Cross Country Tutorial** by John Rafferty  
**Creating and Using Amiga Workbench Icons** by C. Hansel  
**AmigaDOS version 1.2** by Clifford Kent  
**The Amazing MIDI Interface** build your own by Richard Rae  
**AmigaDOS Operating System Calls and**  
**Disk File Management** by D. Haynie  
**Working with the Workbench** by Louis A. Mamakos Prog in C

### Volume 2 Number 3

**The Amiga 2000™** by J. Foust  
A first look at the new, high end Amiga™  
**The Amiga 500™** by John Foust  
A look at the new, low priced Amiga  
**An Analysis of the New Amiga PCs** by J. Foust  
**Speculation on the New Amigas**  
**Gemini** Part II by Jim Meadows  
The concluding article on two-player games  
**Subscribers and Superstars in AmigaBASIC** by Ivan C. Smith  
**The Winter Consumer Electronics Show** by John Foust  
**AmigaTrix** by W. Block Amiga™ shorts  
**Intuition Gadgets** by Harriet Maybeck Tolly  
A journey through gadgetland, using C  
**Shanghai** reviewed by Keith M. Conforti  
**Chessmaster 2000 & Chessmate** reviewed by Edwin V. Apel, Jr.  
**Zing!** from Meridian Software reviewed by Ed Bercoy  
**Fort!** by Jon Bryan Get stereo sound into your Fortn programs.  
**Assembly Language on the Amiga™** by Chris Martin  
**Roomers** by the Sandito Genlocks are finally shipping, & MORE!!!  
**AmigaNotes** by R. Rae Hum Busters... "No stereo? Y no?..."  
**The AMICUS Network** by J. Foust  
CES, user group issues and Amiga Expo™

### Volume 2 Number 4 1987

**Amazing Interviews** Jim Sachs by S. Hull Amiga Artist  
**The Mouse That Got Restored** by Jerry Hull and Bob Rhode  
**Sluething Public Domain Disks with CLI** by John Foust  
**Highlights: the San Francisco Commodore Show** by S. Hull  
**Speaker Sessions: San Francisco Commodore Show** H. Tolly  
**Household Inventory System in AmigaBASIC™** by B. Catley  
**Secrets of Screen Dumps** by Naskun Okun  
**Using Function Keys with MicroEmacs** by Greg Douglas  
**AmigaTrix** by Warren Block More Amiga shorts  
**Basic Gadgets** by Brian Catley Create gadget functions  
**Gridiron** reviewed by K. Conforti Real football for the Amiga  
**Star Fleet I Version 2.1** reviewed by J. Tracy AmigaSpace  
**The TIC** reviewed by J. Foust Battery powered Clock Calendar  
**Metascope** reviewed by H. Tolly An easy-to-use debugger

### Volume 2 Number 5 1987

**The Perfect Sound Digitizer** review by R. Battle  
**The Future Sound Digitizer** by W. Block Applied Vision's SD  
**Fort!** by J. Bryan comparing J-Forth and Multi-Forth.  
**Basic Input** by B. Catley AmigaBASIC input routine for use in  
all your programs.  
**Writing a SoundScope Module in C** by T. Fay Programming with  
MIDI, Amiga and SoundScope by SoundScope author.

### Volume 2 Number 5 1987 continued

**Programming in 68000 Assembly Language** by C. Martin  
Continuing with Counters & Addressing Modes.  
**Using FutureSound with AmigaBASIC** by J. Meadows  
**AmigaBASIC Programming** with real, digitized STEREO  
**AmigaNotes** Rich Rae reviews SoundScope Sound Sampler.  
**More AmigaNotes** by R. Rae A further look at Perfect Sound.  
**Waveform Workshop in AmigaBASIC** by J. Shields edit & save  
waveform for use in other AmigaBASIC programs.  
**The Mimetics Pro MIDI Studio** by Sullivan, Jeffery  
A review of Mimetics' music editor/player.  
**Intuition Gadgets Part II** by H. Maybeck Tolly Boolean gadgets  
provide the user with an on/off user interface.

### Volume 2 Number 6 1987

**Fort!** by J. Bryan Access resources in the ROM kernel.  
**The Amazing Computing Hard Disk Review** by J. Foust & S.  
Leemon In-depth looks at the LHD, Hard Drive, Microbots™  
**MAS-Drive20**, Byte by Byte's PAL Jr., Supra's 4x4 Hard Drive and  
Xebec's 9720H Hard Drive. Also, a look at disk driver software  
currently under development.  
**Modula-2 AmigaDOS™ Utilities** by S. Fawciszewski A  
Calls to AmigaDOS and the ROM kernel.  
**Amiga Expansion Peripheral** by J. Foust  
Explanation of Amiga expansion peripherals.  
**Amiga Technical Support** by J. Foust  
How and where to get Amiga tech support.  
**Goodbye Los Gatos** by J. Foust Cooing Los Gatos.  
**The Amicus Network** by J. Foust West Coast Computer Faire.  
**Metascope Shell and Toolkit** by J. Foust A review  
**The Magic Sac** by J. Foust Run Mac programs on your Amiga.  
**What You Should Know Before Choosing an Amiga 1000**  
Expansion Device by S. Grant  
**7 Assemblers for the Amiga** by G. Hull Choose your assembler  
**Shake Up Replaces Top Management** at Commodore by S. Hull  
**Peter J. Baczorby** S. Hull Manager at CBM gives an inside look  
**Logitrix** A review by Richard Knipper  
**Organize!** by a review Richard Knipper database.  
**68000 Assembly Language Programming on the Amiga**  
by Chris Martin  
**Superbase Personal Relational Database** by Ray McCabe  
**AmigaNotes** by Rae, Richard A look at FutureSound  
**Commodore Shows the Amiga 2000 and 500** at the Boston  
Computer Society by H. Maybeck Tolly

### Volume 2, Number 7 1987

**New Breed of Video Products** by John Foust...  
**Very Wild!** by Tim Grantham...  
**Video and Your Amiga** by Oran Sands III  
**Amigas & Weather Forecasting** by Brenden Larson  
**A-Squared and the Live! Video Digitizer** by John Foust.  
**Aegis Animator Scripts and Cel Animation** by John Foust  
**Quality Video from a Quality Computer** by Oran Sands III.  
**Is IFF Really a Standard?** by John Foust.  
**Amazing Stories and the Amiga™** by John Foust.  
**All about Printer Drivers** by Richard Blakak  
**Intuition Gadgets** by Harriet Maybeck Tolly.  
**Deluxe Video 1.2** by Bob Eller  
**Pro Video CG1** by Oran Sands III.  
**Digi-View 2.0 Digitizer/Software** by Jennifer M. Janik  
**Prism HAM Editor** from Impulse by Jennifer M. Janik  
**Easy! drawing tablet** by John Foust.  
**ESA's Turbo-Amiga Tower** by Alfred Aburto  
**68000 Assembly Language** by Chris Martin.



## Volume 2, Number 8 1987

This month *Amazing Computing™* focuses on entertainment packages for the Amiga. Amazing game reviews...

SDI, Earl Weaver Baseball, Portal, The Surgeon, Little Computer People, Sindbad, StarGlider, King's Quest II and III, Faery Tale Adventure, Ultima III, Facets of Adventure, Video Vegas and Bard's Tale.

Plus *Amazing* monthly columns... Amiga Notes, Roomers, Modula-2, 68000 Assembly Language and The Amicus Network. Disk-2-Disk by Matthew Leeds. The ColorFonts Standard by John Foust. Skinny C Programs by Robert Riemersma, Jr. Hidden Messages in Your Amiga™ by John Foust. The Consumer Electronics Show and Comdex by J. Foust.

## Volume 2, Number 9 1987

Analyze 2.0 reviewed by Kim Schaffer. Impact Business Graphics review by Chuck Raudonis. Microfine Filter review by Harv Laser. PageSetter review by Rick Wirth. Gizmoz Productivity Set 2.0 reviewed by Bob Eller. Kickwork review by Harv Laser. Diga Telecommunications Package review by Steve Hull. Mouse Time and Timesaver review by John Foust. Insider Memory Expansion review by James O'Keane. Microbots Starboard-2 review by S. Fawciszewski. Leather Goddesses of Phobos by Harriet Maybeck-Tolly. Lattice C Compiler Version 3.10 reviewed by Gary Sarff. Manx 3.4a Update reviewed by John Foust. AC-BASIC reviewed by Sheldon Leemon. AC-BASIC Compiler an alternative comparison by B. Catley. Modula-2 Programming S. Fawciszewski Raw Console Dev. Events Directory Listings Under AmigaDOS by Dave Haynie. AmigaBASIC Patterns by Brian Catley. Programming with SoundScale Todor Fay manipulate's samples. Bill Volk, Vice-President Aegis Development, by Steve Hull. Jim Goodnow, Developer of Manx C™ interview by Harriet M. Tolly. Plus a great collection of monthly columns...

## Volume 2, Number 10 1987

Max Headroom and the Amiga by John Foust. Taking the Perfect Screen Shot by Keith Conliff. Amiga Artist: Brian Williams by John Foust. Amiga Forum on CompuServe™... Software Publishing Conference Transcript by Richard Rae. All About Online Conferencing by Richard Rae. dBMAN reviewed by Clifford Kent. Amiga Pascal reviewed by Michael McNeil. AC-BASIC Compiler reviewed by Brian Catley. 68000 Assembly Language by Chris Martin. Amiga Programming: Amiga BASIC Structures by Steve Michel. Quick and Dirty Bobs by Michael Swinger. Directory Listings Under Amiga-DOS, Part II by Dave Haynie. Fast File I/O with Modula-2 by Steve Fawciszewski. Window I/O by Read Predmore. Plus a great collection of monthly columns...

## Volume 2, Number 11 1987

Word Processors Rundown by Geoff Gamble. ProWrite, Scribble, and WordPerfect compared. LPD Writer Review by Marion Deland. Aedil Review by Harv Laser. WordPerfect Review by Harv Laser. Jaz: Jan Interview by Ed Bercozitz—StarGlider author speaks! Do-it-yourself Improvements to the Amiga Genlock. Digi-Paint Review by Harv Laser. Script 3D Review by Steve Pietrowicz. Shadowgate Review by Linda Kaplan. TeleGames Review by Michael T. Cabral. Reason Review: an intense grammar examination application. As I See It by Eddie Churchill. WordPerfect, Gizmoz V2.0 and Zing! AmigaNotes by R. Rae 4 electronic music books. Modula-2 Programming by S. Fawciszewski devices, I/O, serial port. 68000 Assembly Language by Chris Martin Display routines. The AMICUS Network by John Foust—Desktop Publishing, Seybold C Animation Part II by Mike Swinger Animation Objects. BASIC Text by Brian Catley Pixel perfect text positioning. SoundScale Part III by Todor Fay VU Meter and more. Fun with Amiga Numbers by Alan Barnett. File Browser by Bryan Catley—Full Feature BASIC File Browsing. Plus a great collection of monthly columns...

## Volume 2, Number 12 1987

The Ultimate Video Accessory by Larry White. The Sony Connection by Stewart Cobb. 15-Puzzle in AmigaBASIC by Zoltan Szepel. Life, Part I: The Beginning by Gerald Hull. The ultra-complex nine bit solution to the "Game of Life." Amiga Virus! by C. Paul Castonguay. CLI Arguments in Adapter by Barry Massoni. MIDI Interface Adapter by C. Paul Castonguay. Amiga 1000-style MIDI interfaces can fit A2000s or 500s. Modula-2 by S. Fawciszewski Part I: command line calculator. AmigaNotes by R. Rae audio changes made in the A500 & A2000. Animation for C Rookies: Part III by M. Swinger double-buffering. The Big Picture by Stephen R. Pietrowicz. Karate Kid Review by John Foust, James O'Keane, and Rick Wirth. Three C-64 experts investigate a new Amiga 64 emulator. A-Talk-Plus Review by Brendan Larson. Calligrapher Review by John Foust. Animator: Apprentice Review by John Foust. Playing Dynamic Drums on the Amiga by David N. Blank. WordPerfect Review by Steve Hull. Insider/Kwikstart Review by Ernest P. Viveiros Sr. RAM & ROM expansion: Comments and installation tips. Forth by Jon Bryan DurrPorth utility for your Multi-Forth toolbox. As I See It by Eddie Churchill Digi-Paint, Portal, and Videocase 3D. The Commodore Show and AmigaExpo: New York! Plus a great collection of monthly columns...

## Volume 3, Number 1 1988

AmigaNotes by Richard Rae Amiga digital music generation. C Animation Part IV by Michael Swinger. Forth by John Bryan Sorting out Amiga CHIP and FAST memory. The Big Picture by Warren Ring Daring assembler language programming: CLI system calls and manipulating disk files. 68000 Assembly Language Programming by Chris Martin "Create a multi-color screen without using Intuition routines!" Modula-2 Programming by S. Fawciszewski A new modula-2! Amicus Network Special Report: Fall COMDEX by J. Foust. The Ultimate Video Accessory: Part II by Larry White. Life: Part II by Gerald Hull The Amiga blitter. FormatMaster: Professional Disk Formatting Engine by C. Mann Put Batch language to work on the drudgery of disk formatting. BSpread by Brian Catley full featured AmigaBASIC spreadsheet. AmigaForum Transcript ed. by Rick Rae Amiga's Dave Haynie. Halcac Review by Chuck Raudonis easy to use, spreadsheet. VIP Professional Review by S. Mitchell Manage stock portfolio. Money Mentor Review by S. Kemp Personal finance system. Investor's Advantage Review by Richard Knepper plus "Poor Man's Guide to the Stock Market." Plus a great collection of monthly columns...

## Volume 3, Number 2 1988

Laser Light Shows with the Amiga by Patrick Murphy Lights and the Amiga: A Dazzling Tandem. The Ultimate Video Accessory: Part III by Larry White. Take the final step toward designing your own videos. Our First Desktop Video by Larry White. Step-by-step guide to organizing & presenting your Amiga video. Hooked on the Amiga with Fred Fish interview by Ed Bercozitz. Photo Quality Reproduction with the Amiga and Digi-View by Stephen Lebas. Balancing your Checkbook with WordPerfect Macros by S. Hull Hand your checkbook worries over to the Amiga. More Basic Text by Bryan Catley easier text on an Amiga screen. Life: Part III by Gerald Hull. Sries winds up with famed nine-bit calculation & source to LIFER. Solutions to Linear Algebra through Matrix Computations by Robert Ellis. Simply matrix algebra with basic operations & routines. Modula-2 Programming by Steve Fawciszewski. Catching up with Calc-a source follow-up. 68000 Assembly Language Programming by Chris Martin Graphics: Part II of Assemblage. Arazox's Tomb Interview by Kenneth E. Schaefer. AIRT by S. Fawciszewski innovative icon-based program, lang. Forms in Flight by S. Pietrowicz Render & Animate 3D objects. Silicon Dreams and the Jewel of Darkness by K. E. Schaefer. Leisure suit Larry by Kenneth E. Schaefer. Two New Entries From Microbotics by John Foust. M501 Expansion & Starboard II MultiFunctionBoard. Mindlight 7 and People Meter by John Foust. Phantasia Ken E. Schaefer Amazing Phantasia Character Editor. Plus a great collection of monthly columns...

## Volume 3, Number 3 1988

Desktop Video, Part IV by Larry White. Put all the pieces together—the desktop video commercial. The Hidden Power of CLI Batch File Processing by J. Rothman. Make your Amiga easier to use with CLI Batch files. A Conference with Eric Graham edited by John Foust. The mastermind behind Sculpt 3D and Animate 3D. Perry Kivoliowitz Interviewed by Ed Bercozitz Amiga insights from a major developer and personality. Jean Moebius' Giraud Interviewed by Edward L. Fadigan. Avant-garde art comes to the Amiga in dazzling form. PAL Help by Perry Kivoliowitz A1000 expansion reliability. Boolean Function Minimization by Steven M. Hart. A useful digital design tool in AmigaBASIC. Amiga Serial Port and MIDI Compatibility for Your A2000! by L. Ritter and G. Rentez Add an A1000-style serial port to the A2000! Electric Network Solutions the Matrix Way by Robert Ellis Engineers! Practical routines for using matrix algebra. The A.M.U.G. BBS List compiled by Joe Rothman, Chet Solace, & Dorothy Dean 514 BBS phone numbers in the U.S. & Canada. FACC II reviewed by Graham Kinsey Speed your floppy drives. Uninvited reviewed by K. E. Schaefer. Flow reviewed by Pamela Rothman brainstorming into mental art. Benchmark Modula-2 Compiler reviewed by Richie Bielski. Modula-2 Programming by Steve Fawciszewski. The gameport device and simple sprites in action. AmigaNotes by R. Rae A1000! Software-switchable output filter. Roomers by The Bandito AmigaExpo, Kickstart 1.4, Commodore The Big Picture by Warren Ring—Unified Field Theory! Plus a great collection of monthly columns...

## Volume 3, Number 4 1988

Highlights from AmigaExpo, Los Angeles by Steve Hull. Writing a SoundScale Patch Librarian T. Fay System Exclusive Upgrade Your A1000 to A500/2000 Audio Power—by H. Bassen. Modifications to help your A1000 make sweet music, too! Amiga Audio Guide Listing of all Amiga audio products. Gels in Multi-Forth by John Bushakra. Microbotics by Patrick J. Horgan. Ease the trauma of assembly language programming. Amiga Audio Sources The folks behind all those audio products. Take Five! by Steve Hull five Amiga games reviewed. Amiga Notes by Rick Rae A basic tour of Amiga audio. The Ultimate Video Accessory, Part V by Larry White. Bug Bytes by John Steiner. The Big Picture by Warren Ring Part II Unified Field Theory. Roomers by The Bandito. Hardware hijinx... Toasted video... the dream Amiga... and more! In the Public Domain by C.W. Flatts. Time Bandit review by Keith Conliff. AudioMaster review by B. Larson Real-time digitizing samples. Music Mouse review by J. Henry Lowengard. Making music without lifting a finger from the mouse. Amiga-Talk/Canadian Version review by Ed Bercozitz. A Canadian income tax planning, preparation, & analysis package. SAM BASIC review by Brian Catley. A new BASIC which exploits even more unique Amiga features.

## Volume 3, Number 5 1988

Interactive Startup Sequence by Udo Pernisz. The Command Line Part I by Rich Falconburg. AmigaTrix III by Warren Block—Tips and tidbits to ease Amiga life. Amiga Product Guide: Hardware Edition. Proletariat Programming by P. Quaid—Public domain compilers. The Companion by P. Gosselin Amiga's Event Handling capability. MindLight 7 reviewed by David N. Blank. VideoScope 3-D 2.0 reviewed by David Hopkins. Extend reviewed by Bryan D. Catley—An AmigaBASIC extension. AssemPro reviewed by Stephen Kemp. Opening a door to assembly language programming. APL68000 reviewed by Roger Nelson. Book Reviews by Richard Grace—Three "C" programming texts. CBTree reviewed by Michael Listman. A tidy collection of functions to aid the C programmer. The Big Picture by Warren Ring. The three-part Unified Field Theory winds up. Modula-2 by Steve Fawciszewski. Termination modules for Benchmark and TDI compilers. 68000 Assembly Language by Chris Martin. Peeling away the complication of display routines. Plus a great collection of monthly columns...

## Volume 3, Number 6 1988

Bear Time Reviewed by Steve Carter. What makes this inexpensive A1000 battery-backed clock tick? Acquisition Reviewed by David N. Blank. A look inside the latest release of a powerful relational database. Butcher 2.0 Reviewed by Gerald Hull. A tidy collection of diverse image processing utilities. Reassigning Workbench Disks by John Kennan. Endless disk swapping comes to a merciful end. Product Guide: Software Tools Edition. A listing of all the products you need to put your Amiga to work. An IFF Reader in Multi-Forth by Warren Block. Create an easy to use IFF reader in Multi-Forth. Basic Directory Service Program by Bryan Catley. A programming alternative to the GimmeeZeroZero windows. C Notes from the C Group by Stephen Kemp. A beginner's guide to the power of C programming. An Amiga Forum Conference with Jim Mackraz. The Amiga market as seen by the "Stepfather of Intuition." Son of Seven Assemblers Reviewed by Gerald Hull. A comparative battle between seven native-code assemblers. The 1988 Commodore Amiga Developers Conference. A look inside the conferences held in Washington, D.C. Amiga Working Groups by Perry Kivoliowitz and Eric Lavitsky. An outline of the innovative Amiga Working Groups concept. The Command Line by Rich Falconburg. Exploring the multi-talented LIST command. Plus a great collection of monthly columns...

## Volume 3, Number 7 1988

Look, Up On the Screen, It's an Aml... It's a Pro... It's SuperGen reviewed by Larry White—Genlock comparisons. An Interview with "Anil" Mann, Gary Bonham by B. Larson. An animated conversation with the man behind the format. The Amiga at Spring COMDEX in Atlanta by Ed Bercozitz. Amiga Product Guide: Video/Graphics Edition. Thirteen pages devoted to the Amiga's dazzling strong suit. The Developing Amiga by Steve Pietrowicz. Developers' notes: PD vs. shareware vs. freely distributable. Roll Those Presses! by Barney Schwartz. Welcome to the dandy, demanding world of desktop publishing! Linked Lists in C by W. E. Gammill Put dynamic memory to work! FileGrabber Review by Oran Sands. Capturing an image can now be as fast as punching a single key! A First Look at Interchange reviewed by David Hopkins. Bridge the gap between those incompatible animation packages. Perfect Vision reviewed by Bryan Catley. Capture, digitize and save pictures from any video source. ProWrite 2.0 Review reviewed by Pamela Rothman. A graphic word processor specializing in efficient editing. Doug's Math Quizzes: The Art of Mathematics by R. Bielski. Bear Products MegaHex II Expansion RAM by Steve Carter. The Command Line by Rich Falconburg. Amiga Notes by Rick Rae. The Other Guys' Synthia: A digital synthesizer features won't stop. C Notes from the C Group by Stephen Kemp. Weathering the unknown "C" of basic object and data types. Plus a great collection of monthly columns...

## Volume 3, Number 8 1988

The Command Line by Rich Falconburg. The journey into the CLI continues. The Developing Amiga by Stephen R. Pietrowicz. A gaggle of great programming tools. Modula-2 Programming by Steve Fawciszewski. Libraries and the FFP and IEEE Math Routines. C Notes from the C Group by Stephen Kemp. Arrays and pointers unmasked. Dark Castle reviewed by Keith Conliff—The Black Knight lurks. Pools of Call reviewed by Julie Landry. Leatherneck reviewed by Michael Creeden-Rambo's not so tough! Capone reviewed by Joyce and Robby Hicks—Light Guns blaze. Casino Fever reviewed by Michael T. Cabral—Vegas on Amiga. Ferrari reviewed by Jeffery Scott Hall—Start your engine. Ankorstar reviewed by Graham Kinsey—"blockbuster" Bronkard by Keith Conliff—black hole trekking. Deluxe Productions reviewed by Harv Laser—Video wizardry. Game Pizzazz by Jeffery Scott Hall—Register your questions here. TrackMouse by Darryl Joyce. Convert a standard Atari trackball into a peppy Amiga TrackMouse. Amiga Interface for Blind Users reviewed by Carl W. Mann. An ingenious interface that opens the Amiga to even more users! Video in the Sunshine State reviewed by Stephen R. Pietrowicz. RGB Video Creations hosts a video unveiling! Amiga Product Guide: Games Edition. Tumblin' Tots by David Ashley—assembly language program. Plus a great collection of monthly columns...

## Volume 3, Number 9 1988

The Video Tapes by John Dandurand. A Georgia elementary school puts desktop video to work. Speeding Up Your System by Tony Preston floppy disk caching. Amiga Product Guide: Education Edition. Everything you need to send your Amiga to the head of the class. Computer Aided Instruction by Paul Castonguay. Authoring system in AmigaBASIC. Gels in Multi-Forth, Part II: Screenplay by John Bushakra. Make the IFF converter from Part I easy to use—gadgets, menus, etc. AmigaExpo Midwest '88 by Michael T. Cabral. After taking the coasts by storm, the Amiga wows Chicago. IntelliType by Harv Laser—Learning to type made easy...and fun? Shakespeare by Barney Schwartz—Desktop publishing in full color. XSpecs 3D by Steve Hull—A new dimension in Amiga graphics. AmigaNotes by Richard Rae—How IFF sound samples are stored? Take Five! by Steve Hull—Beat the back-to-school blues! The Command Line by Rich Falconburg—continuing tour of CLI. Hot on the Shelves by Michael T. Cabral & Michael Creeden. What do you get when you combine intense war strategy with a monochrome monitor and desktop presentation? Check it out. Bug Bytes by John Steiner. C Notes from the C Group by Stephen Kemp. Operators, expressions, and statements in C uncovered. Roomers by The Bandito Can Apple IIGS Plus keep Amiga away?

## Volume 3, Number 10 1988

A First Look At: Deluxe PhotoLab reviewed by David Duberman. A paint package, poster-maker, and image processing program. DiskMaster reviewed by Steve Hull—file management utility. DSM: A MC68000 Disassembler reviewed by Gerald Hull. Looking for easily modifiable, assembler-ready code? FBASIC Language System reviewed by Patrick Quaid. BASIC compiler and development system. Hot on the Shelves by Michael T. Cabral—Deviant dogs, gripping gray scales, color cartography, mauling monomers, and much more. The Command Line by Rich Falconburg. NEWCL: A painless way to create a new console window. The Developing Amiga by S. Pietrowicz Usernet—24-Hour News. C Notes from the C Group by Stephen Kemp—loops. Roomers by The Bandito WP wars, ignominious interfaces, & more. PD Serendipity by C.W. Flatts—Fred Fish collections passes 150. Comparison of MultiScan Monitors by Steven Bender. Five multiscan alternatives square off on the desktop. Record Keeping for Free-lancers: A Superbase Professional Tutorial by Marion Deland. Record keeping system for free-lance photographers and others. On the Crafting of Programs by David J. Hankins—A look at optimization kicks off a series of articles on programming savvy. Bob and Ray Meet Frankensteins by Robert D. Asto—Create, animate, and metamorphose graphics objects in AmigaBASIC. Digital Signal Processing in AmigaBASIC by Robert Ellis. Perform your own digital experiments with Fast Fourier Transforms. HAM & AmigaBASIC by Steve Carter—Pack your AmigaBASIC programs with many of the Amiga's 4096 shades! CAI—Computer Aided Instruction: Part II by Paul Castonguay. The Editor program wraps up our authoring system in AmigaBASIC.

## Volume 3, Number 11 1988

Desktop Publishing with Professional Page by Barney Schwartz. Tutorial in document creation, plus some jazzy enhancements. Game Pizzazz by J. Hall gaming hints, tips, high-score secrets. Structures in C by Paul Castonguay C programming in a nutshell. On the Crafting of Programs by D. Hankins speed up your prog. Desktop Video VI: Adding the Third Dimension by Larry White. Unraveling the complexity of 3D for your video creations. A2000 Hard Drive Round Up by Sheldon Leemon. Keyclick by Mike M. Duppung a typewriter click in your keyboard. More Linked Lists in C: Techniques and Applications by Forest W. Arnold Procedures for managing lists, storing diverse data types in the same list, and putting lists to work in your programs. BASIC Link by Brian Zupke Combine individual routines from your program library to create an executable program. The Developing Amiga by Steven Pietrowicz. A look at mysteries and successes behind efficient beta testing. Modeler 3D Preview reviewed by David Hopkins. A peek inside a new, open-ended 3D package. ArtDraw Graphics Tablet reviewed by Keith Conliff. Artists! Meet the future of Amiga graphics. StarGlider II reviewed by Jeffery Scott Hall. Those imitating Eagles are back for another laser-lashing. WShell reviewed by Lawrence Lichtman CLI substitute. Hot on the Shelves by M. Cabral viruses, music, microfiche mastery. PD Serendipity by C.W. Flatts. Fred Fish disks 149-152. Roomers by The Bandito Golden RAM, 16-bit videogames, CD-I, another HAM skirmish... what could possibly be NeXT?

## Volume 3, Number 12 1988

Hot on the shelves by M. T. Cabral. Graphic adventure, control over Preferences, a Postscript print utility, sequence live action animation, a new deal for user groups and the figure construction set. PD Serendipity by C. W. Flatts. Fred Fish disks 158-162. Bug Bytes by John Steiner. All the latest from the world of bugs and upgrades. Roomers by The Bandito. AmigaExpo, C.D. the latest from Commodore and more. AmigaExpo California by Stephen Kemp. Hot—All the news. EMPIRE reviewed by Stephen Kemp. EMPIRE, the game of conquest, has finally come to the Amiga. Virus Infection Protection (V.I.P.) reviewed by Jeffery Scott Hall. What makes a computer sick and the cure. The Command Line by Rich Falconburg. What to do when the commands of AmigaDOS fail. Converting Patch Librarian Files by Phil Saunders. How to get your sounds from there to here. E.C.T. SampleWare by Tim Hamsingh. The E.C.T. samples contain several gems. The Creation of Don Bluth's Dragon's Lair by Randy Linden. A look behind the scenes.



(Roomers, continued from page 54)  
actually hired people. One company  
reported hiring more than 30 new  
employees.—Ed.]

Why layoffs when the software  
business is growing so much? Well,  
times are tougher for the entertainment  
software companies, particularly the big  
ones. Much of the blame lies with  
Nintendo, which is only delivering about  
20% of the cartridges promised, thus  
cutting down on sales for companies in  
the cartridge business. Some companies  
were not able to ship their Christmas  
titles on time, which hurt their profits.  
Overall, the profitability of these bigger  
software companies is lower than they  
wanted, so out comes the ax. Some  
insiders are beginning to suspect that the  
big entertainment software companies  
have reached a plateau where they'll find  
it hard to grow at the rate they've been  
used to. Doing ten or a hundred quality  
software titles in a year is not just a  
matter of scaling up the number of  
people and resources needed for one  
title a year. So the software giants need  
to find a method that works, and until  
they do, they'll have to keep a tight rein  
on spending.

### Creative budget-cutting

In search of such a method,  
Electronic Arts has reorganized once  
again, this time eliminating (see Ed. Note)  
the Creativity Products division they  
invented a couple of years back. In the  
process, they fired most of the creativity  
development people. Those in the know  
say that EA has never really understood  
how to market their creativity software.  
(Have you ever seen any advertising for  
any of their Amiga titles? Well, neither  
has anyone else.) While they continue  
development efforts on programs like  
DeluxePaint III and DeluxeVideo II, it's  
likely that there will be no advertising for  
these titles.

[In reference to Electronic Arts  
eliminating the Creativity Department,  
Electronic Arts President Trip Hawkins  
said that a few years ago, EA had divided  
the Creativity department into three  
divisions. When EA recently decided that  
this structure did not work as expected,  
they re consolidated the three divisions  
into the Electronic Arts Studio. It appears,  
then, that EA has done some creative  
restructuring—not creative budget-  
cutting.]

And don't bet on any new Amiga  
creativity titles being started. But  
DeluxePaint III should ship in March,  
and it looks like a solid hit, even without  
any advertising. Ironically, it's been the  
success of EA's Amiga titles (supposedly  
over 100,000 copies of DeluxePaint  
Amiga have been sold) that prompted  
them to start doing Apple IIGS, IBM, and  
Macintosh creativity software. While  
none of the software for other CPU's has  
ever earned as much as their Amiga  
titles, it's the Amiga product  
development that's getting stopped,  
while they spend their money on IBM  
and Macintosh titles. Go figure.

Mediagenic's acquisition of a major  
Amiga developer has fallen through.  
Apparently, after a close review of their  
company, Mediagenic decided that the  
target's product sales are flagging, no  
new hits are appearing, and the target  
company management seems a little  
bewildered. Besides, Mediagenic has  
spent too much lately anyway on  
acquisitions, and needs time to  
consolidate their gains.

[Mediagenic denied all charges of  
the attempted buyout.—Ed.]

EA just completed another round of  
venture capital financing, adding \$4  
million to their corporate coffers (Trip  
Hawkins said the figure was \$3  
million.—Ed.) This serves as a war chest  
against the \$8 million or so that  
Mediagenic has saved from their salad  
days. The money will probably be used  
by both sides to strengthen their  
entertainment software production and to  
shore up their overseas companies,  
which still aren't showing much vigor.

The one bright spot for the big  
companies is their distribution business,  
which has been booming. The small  
publishers who are affiliated labels,  
letting the big companies distribute their  
products, find that their hot titles get to  
an even wider audience than before.  
And the small publishers seem to be  
much better at turning out hot titles with  
consistency. Look for more of these  
distribution deals in 1989, especially in  
the Amiga market. You may have  
noticed a flood of new software  
companies selling Amiga versions of  
arcade hits, and just new Amiga games in  
general. It's a good bet that many of  
these companies will find it more  
lucrative to let a big software company  
distribute their software than it is to try  
and build their own distribution network.

(continued)

# Amazing COMPUTING

## Expanding Reference

### Volume 3 Number 12 1988 continued

Easy Menus in JForth by Phil Burk  
HELLO WORLD!  
Extending AmigaBasic by John Kennan  
The use of library calls from within AmigaBASIC.  
Better Dead Than Alien reviewed by Jeffery Scott Hall  
Don't fire until you see the greys of their eyes.  
Getting Started in Assembly by Jeff Glatt  
An introduction to Amiga assembly language programming  
AC/BASIC 1.3 reviewed by Bryan Catley  
Release 1.3 of Absoft's AC/BASIC compiler for the Amiga.  
Thexder reviewed by Bruce Jordan  
Thexder turns out to be a real screamer: Action, Adventure,  
Fantastic Sound, and stunning Graphics.  
Magellan: The AMIGA Gets Smart reviewed by Steve Gillmor  
The worlds of artificial intelligence comes to the AMIGA in the form of  
A.I. system software.  
C Notes From The C Group by Stephen Kemp  
Program or function control coding: the case history.  
AmigaDos, Assembly Language, And FileNotes by Dan Huft  
Weapons in the war against file overload; accurate, descriptive file naming.

### Volume 4 Number 1 1989

The Wonderful World of Hashniqua reviewed by Shamus Morter  
A review of the Amiga software products of Hash Enterprises  
Desktop Video by Richard Starr  
Thinking about getting into Video? Here's what you'll need to know.  
Industrial Strength Menus by Robert D'Asio  
Add some snazzy submenus to your AmigaBASIC cuisine  
Second Generation 2D Animation Software by Geoffrey Williams  
Cel Animators and Key Frame Animators,  
how they differ and a look into their use.  
What's The Diff? reviewed by Gerald Hull  
A review of Lattice's Compiler Companion  
Scrolling Through SuperBitMap Windows by Read Predmore  
How to implement SuperBitMaps  
for viewing or drawing into large graphic areas.  
Alive In 3D by Shamus Morter  
A review of Caligari, a High-End 3D sculpting & animation package.  
Sync Tips by Oran J. Sands III  
Dot crawl, the Amiga and composite video devices.  
How May I Animate Thee?, Let Me Count The Ways—  
by Shamus Morter  
An overview of animation techniques.  
Stop-Motion Animation On The Amiga by Brian Zupke  
A hands on approach to animation and the Amiga.  
Roomers by The Bandito  
Commodore's deal, RAM chip crisis, and more!  
C Notes From The C Group by Stephen Kemp  
Structures - A powerful feature of C  
On the Crafting of Programs by David J. Hankins  
What Format is right for you  
The Command Line by Rich Falconburg  
A look at new and improved Assembly Language commands  
Questron II reviewed by Jeffery Scott Hall  
Questron II - It's a journey back in time  
Pointers, Function Pointers, and Pointer Declarations in C  
by Forest W. Arnold  
Reducing data type dependencies  
Las Vegas Comdex Report by Louise Brinkmann  
Commodore's new 2500, 2500 UX, and more!  
Philadelphia World of Commodore by Chris Darsch & Rick Rae  
Highlights of Philadelphia's Commodore Show  
STELLARYX Review by Stephen Kemp  
Exciting & challenging! Terrific stereo and sound effects  
Arkanoïd Imposters: Unmasking the impostors. reviewed by Jeffery Scott Hall  
A look at Arkanoïd look-alikes  
Bug Bytes by John Steiner  
Bugs and upgrades  
Death of a Process by Mark Cashman  
Develop an error handling module in Modula-2

To be continued.....

To Order  
Back Issues,  
please use the order  
form on page CIII



**S-VIEW**  
THE AMIGA TO S-VIDEO CONNECTION

**Assistant  
Director**  
A tutorial for "The Director" which includes a voice feature..

**inky dink**  
Ink for Xerox and Diablo ink-jet printers  
4 colors, 3 sizes  
HALF THE PRICE OF THE  
MANUFACTURER'S INK

**Brummbaer's Backgrounds**  
COMING SOON

**SOFTWARE SENSATIONS**  
Los Angeles

1441 So. Robertson Bl. Los Angeles, Ca. 90035 (213) 277-8272

According to several industry sources, the Apple IIGS is about equivalent to the Amiga in terms of software sales per title in the US. This must be disappointing for Apple, which has spent many times Commodore's advertising and marketing budget on promoting the IIGS. Maybe someone should point out to Apple that for the price of a basic IIGS system, you can get an A2000 with a hard drive and an accelerator, plus a good printer and a selection of software. You have to feel sorry for the poor yuppies that buy a IIGS because they don't know any better...

The Bandito has heard that the first shipment of DeluxePrint II has a *leettle* problem with Workbench 1.3 — it doesn't work too well. If you have DeluxePrint II (so you're the one!) call EA customer service and they'll get it fixed for you. Did anyone test DeluxePrint II with 1.3? Some mysteries will never be solved.

#### Hot new games

Hot new titles include *Who Framed Roger Rabbit*, *Dragon's Lair*, and *The Dungeonmaster*. The music on some of these games is getting so good the Bandito is humming the tunes just for fun. (Maybe someone should put some of these tunes in IFF format on a disk and sell them through K-Tel Records.) The trend in games is toward real-time action, and bigger figures on screen. We're seeing the rise of the 16-bit game—the difference between titles designed for the 16-bit machines and the 8-bit machines is becoming more apparent. Even the better 8-bit titles come from 16-bit game designs. *Bard's Tale* originated on an Apple II, whereas *Dungeonmaster* is an Amiga title, and you can see the difference in the real-time action.

#### DVI news

After Intel bought all of the Digital Video Interactive technology and hired the 25 engineer team responsible, they said that other companies, possibly even IBM, will announce DVI-based products

in 1989. They've launched a two-year program to engineer a low-cost DVI chip set, which they intend to make an industry standard. Could future Amigas get a hold of this? It seems like a much more natural match than an IBM, after all. The Bandito would hate to have to buy an IBM to get the latest in video technology. How about it, Commodore?

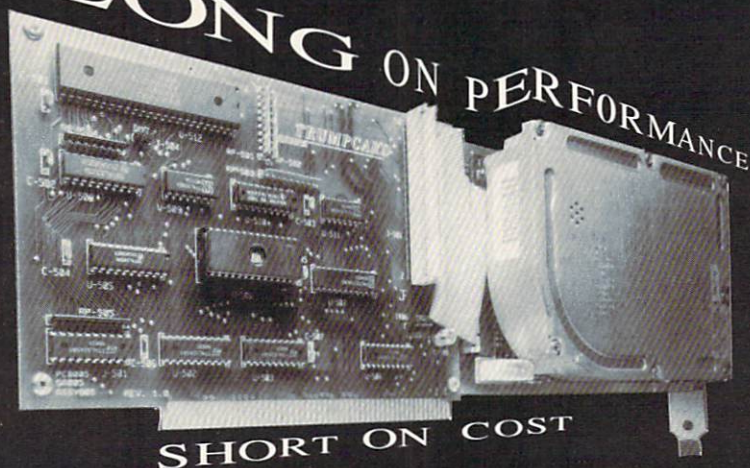
One early Amiga developer has been strangely silent the past six months or so, keeping away from most advertising and public appearances. Unknown to many outside the developer community, the owners of the company are firm believers in trance channeling, and this has guided some of their business decisions. Perhaps RAM-tha, spirit guide of ancient Atlantis, has been advising them lately to take a low profile. ("Stay away from trade shows and advertise not, lest your inventory shortages be exposed.") The Bandito thinks that they should stay away from spirits and get some capital together to keep their products in stock.

Amiga have been strong over Christmas, and the VCR bundle seems to have helped. The VCR offered is a plain-jane model, but it does the job. The funny part is that there's no way in the bundle to hook the VCR up to the Amiga 500, since there's no composite video output, so it's rather hard for the new Amiga buyer to understand what connection a VCR has with an Amiga, if you get the pun. You have to buy an adapter to hook up the two—wouldn't it have made more sense to bundle the adapter with the package? Word is that Commodore got a really great deal on these VCRs as a manufacturer's closeout, so the Bandito guesses that they just didn't care about the technical details of hardware interfacing.

One Amiga dealer has found a great way to sell Amigas to video professionals — don't tell them they're buying an Amiga. He advertises a character generator for video work, showing titles created with JDK Images' Pro Video CG1. He doesn't mention the software or the fact that it's bundled with an Amiga 2000 system anywhere in his advertising. But what's the difference? It's cheaper than most character generators on the market, easier to use, and a hell of a lot more flexible. Commodore would probably prefer that he mention the Amiga, but then again Commodore probably isn't even aware



# TRUMPCARD DISK CONTROLLERS ARE...



- Loads DPAINT 2 in 1.4 seconds, identical speed as DMA to cache type controllers costing HUNDREDS MORE! \*
- Exclusive *IVS SMARTBOOT* autoboots ALL drives (including Seagate) from cold start
- Up to 7 SCSI drives can be daisy chained to one Trumpcard
- Full support of Logical Units with Adaptec controllers
- Supports all Quantum, Seagate and Miniscribe drives
- User customizable driver supports virtually any SCSI hard disk drive. The list includes over 15 devices and is growing
- Half length card takes only 1/2 slot
- Optional brackets mount 3.5" drives to card (\$24.95)
- Exclusive *IVS* checklist configuration software makes hard drive setup a breeze
- 2 ft. 50 pin SCSI cable included with every Trumpcard
- Full 1 year warranty parts and labor

\* Dpaint 2 loaded from Quantum Prodrive 40 with AmigaDOS 1.3 FastFileSystem

DPAINT 2 is a trademark of Electronic Arts  
AmigaDOS is a trademark of Commodore Business Machines

**LIST PRICE: \$199.95!**  
AUTOBOOT ROM INCLUDED

*Why Spend More??*



## TRUMPCARD

15201 SANTA GERTRUDES AVE. STE. Y102 LA MIRADA, CA. 90638 PHONE: (714) 994-4443

that this guy exists. But he's providing a great deal to the video pros — what other character generator can you play games on?

Atari Games (not the same as Atari Computers) has filed suit against Nintendo, for restraint of trade, moper and doper, etc., charging that Nintendo's proprietary attitude toward their game cartridges is illegal. Atari Games is asking for \$35 million and is seeking triple damages under antitrust law, which would make the total take over \$100 million. To prove that they're serious, Atari Games has also announced that they are producing their own Nintendo-compatible cartridges, which is the first time that anybody has had the *cojones* to challenge Nintendo's legal web of patents, copyrights, and trademarks. Companies like Mediagenic are watching this with intense interest; they'd love to be able to manufacture their own cartridges without having to go through Nintendo. This action could

affect the profits at many software companies, though the litigation probably will take years to resolve.

Well, it is about time for the Bandito to go into a trance and predict what 1989 holds. No promises about how good these predictions are, except that they should beat out the National Enquirer's accuracy by an order of magnitude. So without further ado, here are the Bandito's 1989 predictions:

- The number of Amigas sold will reach 2 million worldwide by the end of the year.
- More games will appear in the coming year than the total of Amiga games already released.
- Kickstart 1.4 will not appear until the end of the year.
- The Amiga 3000 will not appear at all in 1989.

• Commodore will announce a CD-ROM drive under development.

• The Amiga 500 will be sold in Toys R Us by the end of the year.

• Atari will announce four new computers and ship none of them.

• Commodore will announce four new computers and ship three of them.

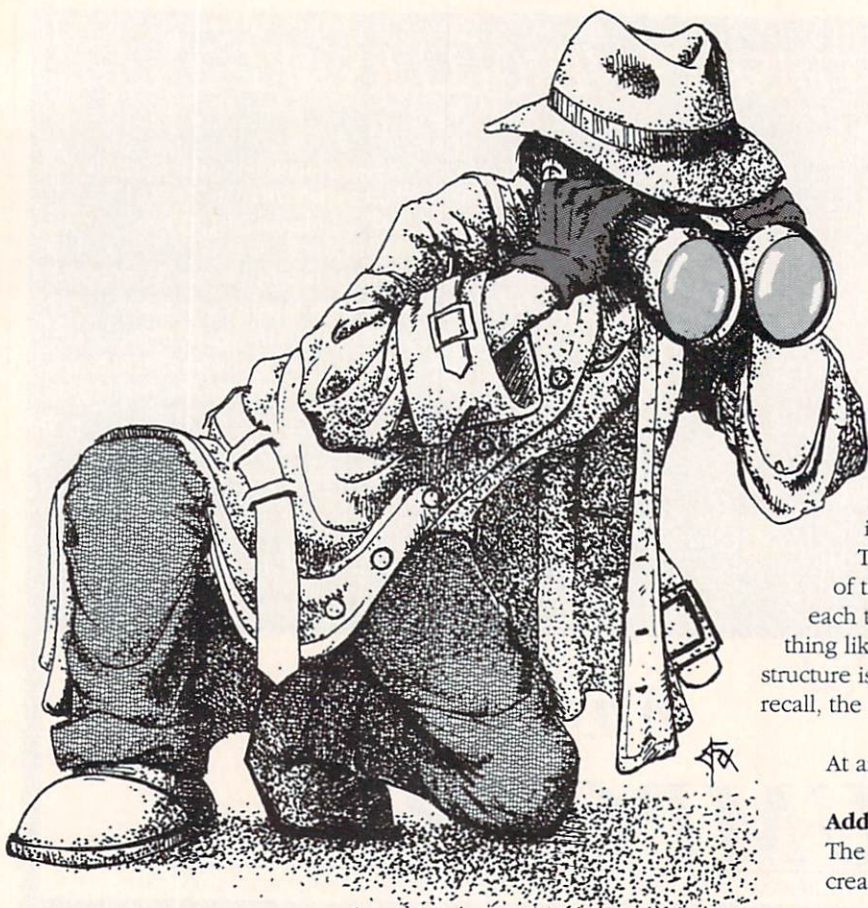
• The Amiga will get some great national PR this year in major media, but it won't be because of Commodore's efforts.

• The C64 will finally begin to succumb to the pressure of low-priced IBM clones, and sales will fade strongly, with the final closeouts being Christmas 1989.

• The Bandito's identity will not be disclosed, but both John Dvorak and Geraldo Rivera will be investigated as prime suspects.

•AC•





### **Reasons for this program**

Have you ever run a program on the Amiga that seemed to be out to lunch? Did you wish for a tool that would allow you to see if the program was working, or whether it was simply hung? Well, wish no more. Here is such a device. Called "Spy", the utility will display the state of task as well as some other information, like the contents of the CPU's registers and the task name.

### **Multitasking and the Exec**

Multitasking on the Amiga is principally the same as multitasking on other real-time systems. The CPU executes a task until one of three things happens: either the task's time slice (the amount of time allotted to the task for execution) expires, another higher priority task becomes ready to be executed, or the running task decides to voluntarily give up control of the CPU. (This also happens when the task needs to do some I/O which is handled by something other than the CPU).

When a task loses control of the CPU, the CPU saves all the task's pertinent information somewhere in memory so when the task gets to run again, it will be able to resume where it left off. The CPU then restores the pertinent information of the other task (which is about to execute). This pertinent information is called the task "context," and the act of switching from one task to another is called a "context switch". On the Amiga, the task context consists of all the CPU registers: 8 data registers, 7 address registers, a stack pointer, a status register, and a program counter. Right before a task is "switched out," all the CPU registers are saved on the task's stack. Right before a task

is switched in, all the CPU registers are loaded from the task's stack. When a task resumes execution, it doesn't "know" that it has lost control of the CPU. As far as it's concerned, everything is exactly the way it was the moment before.

The task context is only relevant to retaining the state of the CPU for the task. Other information significant to each task is stored in a data structure, usually called something like a "task control block." One item kept in this structure is a pointer to the task's stack (where, as you might recall, the task context can be found).

At any given moment a task can be in one of 6 states:

#### **Added**

The task was just added to the task list (i.e., it was just created).

#### **Running**

The task is currently in control of the CPU.

#### **Ready**

The task is ready to execute, but is not in control of the CPU. The task will get CPU time based on its priority relative to other "ready" tasks in the system.

#### **Waiting**

The task is waiting for some event to occur. This might be a completion of an I/O request, a timer expiring, etc.

#### **Removed**

The task is being removed from the system.

#### **Exception**

The task is scheduled for special exception processing.

The state of a task is also maintained in the task control block.

The Amiga operating system maintains a list of task control records, one for each task running on the system. There are actually two such lists, one for tasks in the "ready" state, and one for tasks in the "waiting" state. Both lists are pointed to by fields in the ExecBase record. (They are called, naturally enough, 'TaskReady' and 'TaskWait'.)



# Spy Snooping on Tasks in Modula-2

by Steve Faiwieszewski

## Theory of Operation

Spy displays a task's state and context by examining the task's control record. To display the context, the 'tcSPReg' field is used to find out where the task's stack pointer is. Then the stack is examined to determine all the register values for the task's context. I couldn't find any clear documentation explaining the order of the register values on the stack, but I've determined what it is through experimentation.

Multitasking must be disabled while Spy looks up the task context on the stack, since it is essential that target task does not get a chance to run. (That would mean that its stack pointer might change, causing Spy to display incorrect values.) In order not to degrade system performance, the time that multitasking is disabled should be kept to a minimum. Therefore, Spy only disables multitasking while collecting the data to display. The actual display is done AFTER multitasking is enabled again.

## Using it

Spy can be run from the CLI or from the Workbench. When run from the CLI with no command line arguments, or when run from the Workbench, Spy will display a list of all the tasks currently found on the system. When the user double clicks on a task's name, Spy will display the task's context, state, and name. The display is updated continuously every tenth of a second. Spy can also be invoked from the CLI with a task address given as an argument on the command line (must be specified in Hex). In such a case no list of tasks is displayed.

## Some Program Details

The proportional gadget used in conjunction with the displayed task list was created using the procedure 'AddGadgetProp' from module 'SimpleGadget'. This module is part of the optional 'Simple' support package available from Avant Garde Software. I've used this module mainly because I was lazy and did not want to set up the code for it. Those who do not have the 'Simple' package, or those who want to make the executable file a bit smaller, can write the necessary code for the prop gadget themselves.

I wanted the displayed list to be updated while the user slides the proportional gadget. There are two ways of doing this. One is to specify 'FollowMouse' in the gadget activation field as well as set the 'MouseMove' IDCMP flag. This would cause Intuition to send mouse movement messages when the slider is moved around. Another way is to detect when the left button is pressed on the slider and then turn on 'IntuiTicks', which causes Intuition to send timed messages every few ticks. While the latter method requires a bit more code, I chose it as it seemed to give better results.

*The program consists of two main modules:*

### Spy

This is the main program module. It is responsible for parsing the command line argument and for displaying the list of all tasks.

### Snoop

This module contains the code for the actual "spying" on the target task.

*Also used are two other modules:*

### Termination

This module was published in a previous article.

### IntuiCommon

This module contains miscellaneous routines which I found useful in dealing with Intuition. It was published as part of the MultiSort article.

## Going further

Spy can be enhanced in a number of ways.

1. Currently, Spy runs in a loop displaying the target task's info every tenth of a second. The "sleeping" is done using the AmigaDOS 'Delay' procedure, which is hardcoded to delay for 5 ticks (a tick is 1/50 of a second). An enhancement can be made to make the delay interval variable, and the timer device can be used to make the timing more precise.

2. When Spy is first invoked with no command line argument, a list of tasks is displayed. That list is static—it gets created once and then just displayed. If new tasks are added to, or old tasks removed from the system, the task list display will not reflect the change. Spy could be improved to provide a dynamic task list display.

3. Currently, only the task name, state and register values are displayed. Other specifics, such as memory allocations, signals received, and signals waited on, can be added to the display.

*(continued)*



## Erich Stein & Associates, Inc.

Public Relations Consultants

*Because the quality of your  
reputation is just as important  
as the quality of your product.*

PO Box 695

Denver, Colorado 80201

TEL (303) 733-3707

### A bug in the Benchmark Tasks Module

While working on Spy, I came across an inaccuracy in Benchmark's 'Tasks' module. The 'tcState' field of the 'Task' record is declared as 'TaskStateSet', where it should be declared as 'TaskState'. The workaround is quite easy. Simply type transfer 'tcState' to 'TaskState', as in the following:

```
state := TaskState(TaskPointer^.tcState);
```

This is exactly what's done in Spy.

### Late breaking Modula-2 News

Phil Camp of M2S announced on CompuServe that M2S will release a new Modula-2 compiler for the Amiga shortly. The new compiler is one-pass (just like Benchmark and M2Amiga), and is integrated with an editor that is supposedly equipped with such desirable features as automatic casing of Modula-2 keywords and identifier completion (so you won't have to type all of 'DrawerDataFileSize').

M2S is the British company responsible for the TDI compiler. TDI is the firm that has distributing rights for that compiler in the U.S. The new compiler should be ready sometime in mid-November.

I hope to review this new compiler in a future article.

DEFINITION MODULE Snoop;

```
(*****  
(* Snoop : The core of the Spy program *)  
(*  
(* Written by Steve Faiwizewski, June 1988 *)  
(*  
(* Not to be used for commercial purpose *)  
*****)
```

```
FROM Tasks IMPORT TaskPtr;  
FROM Intuition IMPORT WindowPtr;
```

CONST

```
WINDOWLEFT = 150;  
WINDOWTOP = 20;  
WIDTH = 300;  
HEIGHT = 150;
```

VAR

```
SpyWindow : WindowPtr;
```

```
PROCEDURE Observe(TargetTask : TaskPtr);  
(* Display important info on the target task. *)  
(* Repeat doing so until user chooses to exit. *)  
  
END Snoop.
```

IMPLEMENTATION MODULE Snoop;

```
(*****  
(* Snoop : The core of the Spy program *)  
(*  
(* Written by Steve Faiwizewski, June 1988 *)  
(*  
(* Not to be used for commercial purpose *)  
*****)
```

```
FROM IntuiCommon IMPORT OpenSimpleWindow;  
FROM Conversions IMPORT ConvStringToNumber, ConvNumberToString;  
FROM TermInOut IMPORT WriteLn, WriteString, WriteCard, Write;  
FROM Strings IMPORT StringLength;  
FROM Tasks IMPORT Task, TaskPtr, CurrentTask, TaskState,  
FindTask, SignalSet, Wait;  
FROM Interrupts IMPORT Forbid, Permit;  
FROM Nodes IMPORT Node, NodePtr, NTPProcess;  
FROM Ports IMPORT MsgPortPtr, GetMsg, ReplyMsg, WaitPort,  
MessagePtr;  
FROM Text IMPORT Text, TextLength;  
FROM Drawing IMPORT Move, Draw, SetAPen, SetBPen,  
WritePixel, RectFill;  
FROM Rasters IMPORT RastPortPtr;  
FROM Intuition IMPORT WindowFlags, WindowFlagsSet,  
IDCMPFlagsSet, IDCMPFlags,  
WindowPtr, CloseWindow,  
SetWindowTitles, IntuiMessagePtr;  
FROM AmigaDOSProcess IMPORT ProcessPtr, Delay;  
FROM AmigaDOSExt IMPORT CommandLineInterfacePtr;  
FROM SYSTEM IMPORT ADDRESS, ADR, WORD, LONGWORD, BYTE,  
TSIZE;
```

CONST

```
MaxStringSize = 26;  
LetterHeight = 9;  
HorizOffs = 9;
```

TYPE

```
LongPtr = POINTER TO LONGCARD;  
WordPtr = POINTER TO WORD;  
  
StringPointer = POINTER TO ARRAY[0..255] OF CHAR;  
CoordRec = RECORD  
X, Y : CARDINAL;  
END;  
RegRec = RECORD  
Value : LONGWORD;  
Name : ARRAY[0..2] OF CHAR;  
Loc : CoordRec;  
END;
```

VAR

```
rPort : RastPortPtr;  
CurLine : CARDINAL;  
Regs : ARRAY[0..14] OF RegRec;  
PcLoc,  
SrLoc,  
StateLoc,  
ProcNameLoc,  
TaskNameLoc : CoordRec;  
WindowTitleString : ARRAY[0..26] OF CHAR;  
TaskIsProcess : BOOLEAN;  
CmdLineLenPtr : POINTER TO BYTE;  
CmdLineStrPtr : StringPointer;
```



```

(* ***** *)
PROCEDURE NewLine;
BEGIN
    INC(CurLine,LetterHeight);
    Move(rPort^,HorizOffs,CurLine);
END NewLine;

(* ***** *)
PROCEDURE ClearLine(X,Y : CARDINAL);
BEGIN
    SetAPen(rPort^,0);
    RectFill(rPort^,X,Y+2-LetterHeight,WIDTH-3,Y);
    SetAPen(rPort^,1);
END ClearLine;

(* ***** *)
(* $D-*) (* all arguments are really passed by reference *)
(* for efficiency *)
PROCEDURE SetupString(Str : ARRAY OF CHAR;
    VAR Coord : CoordRec; NL : BOOLEAN);
VAR length : CARDINAL;
BEGIN
    length := StringLength(Str);
    Text(rPort^,ADR(Str), length);
    Coord.X := TextLength(rPort^,ADR(Str), length);
    Coord.Y := CurLine;
    IF NL THEN NewLine END;
END SetupString;
(*$D*) (* go back to normal parameter passing *)

(* ***** *)
PROCEDURE IsItAprocess(TargetTask : TaskPtr) : BOOLEAN;
(* Check if the target task is also a process. If that *)
(* is so, then get the pointer to the process name. *)
VAR
    pp : ProcessPtr;
    CliPtr : CommandLineInterfacePtr;
BEGIN
    IF CHAR(TargetTask^.tcNode.lnType) = CHAR(NTProcess) THEN
        pp := ProcessPtr(TargetTask);
        IF pp^.prCLI <> NIL THEN
            CliPtr := ADDRESS(LONGCARD(pp^.prCLI)*4D);
            CmdLineLenPtr :=
                ADDRESS(LONGCARD(CliPtr^.cliCommandName)*4D);
            CmdLineStrPtr :=
                ADDRESS(LONGCARD(CmdLineLenPtr)+1D);
            RETURN TRUE
        END;
    END;
    RETURN FALSE
END IsItAprocess;

(* ***** *)
PROCEDURE SetupWindow(TargetTask : TaskPtr);
(* Create the text for all the fields that will be *)
(* displayed. Keep track of their location in the window. *)
CONST
    TaskNameStr = 'Task Name: ';
    ProcNameStr = 'Proc Name: ';
    StateStr = 'State: ';
    PcStr = 'PC: ';
    SrStr = 'SR: ';
    SpaceStr = ' ';
    ColonStr = ':';
VAR
    i,tmp,
    length : CARDINAL;
    DummyLoc : CoordRec;
    HexStr : ARRAY[0..7] OF CHAR;
BEGIN
    WindowTitleString := 'Spying on Task 0x????????';
    ConvNumberToString(HexStr,TargetTask,TRUE,16,8,'0');
    FOR i := 0 TO 7 DO
        WindowTitleString[i+17] := HexStr[i]
    END;
    SetWindowTitles(SpyWindow^,ADR(WindowTitleString),
        ADR('Spy, written by Steve Faiwizewski'));
    rPort := SpyWindow^.RPort;
    CurLine := 17;
    SetAPen(rPort^,0);
    SetBPen(rPort^,0);
    RectFill(rPort^,5,10,WIDTH-3,HEIGHT-2);
    SetAPen(rPort^,1);
    Move(rPort^,HorizOffs,CurLine);
    SetupString(TaskNameStr,TaskNameLoc,TRUE);
    TaskIsProcess := IsItAprocess(TargetTask);
    IF TaskIsProcess THEN
        SetupString(ProcNameStr,ProcNameLoc,TRUE);
    END;
    SetupString(StateStr,StateLoc,TRUE);
    SetupString(PcStr,PcLoc,TRUE);
    SetupString(SrStr,SrLoc,TRUE);
    length := StringLength(SpaceStr);

```

```

    tmp := TextLength(rPort^,ADR(SpaceStr), length);
    FOR i := 0 TO 7 DO
        WITH Regs[i] DO
            SetupString(Name,Loc,FALSE);
            SetupString(ColonStr,DummyLoc,FALSE);
            INC(Loc.X,DummyLoc.X);
        END;
        IF i < 7 THEN
            WITH Regs[i+8] DO
                Text(rPort^,ADR(SpaceStr), length);
                SetupString(Name,Loc,FALSE);
                SetupString(ColonStr,DummyLoc,TRUE);
                Loc.X :=
                    Loc.X + tmp + Regs[i].Loc.X + DummyLoc.X;
            END
        END;
    END;
END SetupWindow;

(* ***** *)
PROCEDURE ShowProcessName;
VAR
    length : CARDINAL;
    CharPtr : StringPointer;
BEGIN
    IF CHAR(CmdLineLenPtr^) = 0C THEN
        CharPtr := ADR('No Command');
        length := 12;
    ELSE
        CharPtr := CmdLineStrPtr;
        length := CARDINAL(CmdLineLenPtr^);
    END;
    IF length > MaxStringSize THEN
        length := MaxStringSize
    END;
    WITH ProcNameLoc DO
        ClearLine(X,Y);
        Move(rPort^,X,Y);
    END;
    Text(rPort^,CharPtr, length);
END ShowProcessName;

(* ***** *)
PROCEDURE ShowTaskName(CharPtr : StringPointer);
VAR
    length : CARDINAL;
BEGIN
    length := StringLength(CharPtr);
    IF length > MaxStringSize THEN
        length := MaxStringSize
    END;
    WITH TaskNameLoc DO
        ClearLine(X,Y);
        Move(rPort^,X,Y);
    END;
    Text(rPort^,CharPtr, length);
END ShowTaskName;

(* ***** *)
PROCEDURE ShowTaskState(tstate : TaskState);
BEGIN
    WITH StateLoc DO
        ClearLine(X,Y);
        Move(rPort^,X,Y);
    END;
    CASE tstate OF
        TInvalid : Text(rPort^,ADR('Invalid '),8) |
        TAdded : Text(rPort^,ADR('Added '),6) |
        TRun : Text(rPort^,ADR('Run '),4) |
        TReady : Text(rPort^,ADR('Ready '),6) |
        TWait : Text(rPort^,ADR('Wait '),5) |
        TExcept : Text(rPort^,ADR('Except '),7) |
        TRemoved : Text(rPort^,ADR('Removed '),8) |
    END;
END ShowTaskState;

(* ***** *)
PROCEDURE Spy(target : TaskPtr);
(* This is the actual code that looks up all the info *)
(* on the given task, and then displays it. *)
VAR
    Str : ARRAY[0..8] OF CHAR;
    mp : IntuiMessagePtr;
    tstate : TaskState;
    Stack : LongPtr;
    stack2 : WordPtr;
    i : CARDINAL;
    stop : BOOLEAN;
    pc,sr : LONGWORD;
    CharPtr : StringPointer;
BEGIN

```

(continued)



```

REPEAT
  stop := FALSE;
(* Get all important info, but first make sure the rug *)
(* doesn't get pulled from under our feet. *)
  Forbid;
  WITH target^ DO
    CharPtr := tcNode.lnName;
    tstate := TaskState(tcState);
    Stack := tcSPReg;
    pc := Stack^;
    stack2 := WordPtr(LONGCARD(Stack) + 4D);
    sr := LONGWORD(stack2^);
    Stack := LongPtr(LONGCARD(Stack) + 6D);
    FOR i := 0 TO 14 DO
      Regs[i].Value := Stack^;
      Stack := LongPtr(LONGCARD(Stack) + 4D);
    END;
  END; (* with *)
  Permit; (* got everything we needed! *)
  ShowTaskName(CharPtr);
  IF TaskIsProcess THEN
    ShowProcessName;
  END;
  ShowTaskState(tstate);
(* Display the Program Counter *)
  Move(rPort^, PcLoc.X, PcLoc.Y);
  ConvNumberToString(Str, pc, FALSE, 16, 8, '0');
  Text(rPort^, ADR(Str), 8);
(* Display the Status Register *)
  Move(rPort^, SrLoc.X, SrLoc.Y);
  ConvNumberToString(Str, sr, FALSE, 16, 4, '0');
  Text(rPort^, ADR(Str), 4);
(* Display all other registers *)
  FOR i := 0 TO 14 DO
    WITH Regs[i] DO
      Move(rPort^, Loc.X, Loc.Y);
      ConvNumberToString(Str, Value, FALSE, 16, 8, '0');
      Text(rPort^, ADR(Str), 8);
    END;
  END;

  mp := GetMsg(SpyWindow^.UserPort^);
  IF mp <> NIL THEN
    stop := CloseWindow IN mp^.Class;
    ReplyMsg(mp);
  END;
  Delay(5);
  UNTIL stop;
END Spy;

(* ***** *)
PROCEDURE Observe(TargetTask : TaskPtr);
(* Display various things about the target task *)
BEGIN
  IF SpyWindow = NIL THEN
    SpyWindow := OpenSimpleWindow(WIDTH, HEIGHT, WINDOWLEFT,
      WINDOWTOP, NIL,
      WindowFlagsSet{WindowDrag, WindowDepth,
        WindowClose, NoCareRefresh},
      IDCMPFlagsSet{CloseWindow}, NIL, NIL);
  END;
  IF SpyWindow = NIL THEN
    WriteString('Could not open window!'); WriteLn
  ELSE
    SetupWindow(TargetTask);
    Spy(TargetTask);
    CloseWindow(SpyWindow^);
    SpyWindow := NIL;
  END;
END Observe;

(* ***** *)
PROCEDURE InitRegNames;
BEGIN
  Regs[0].Name := 'D0';
  Regs[1].Name := 'D1';
  Regs[2].Name := 'D2';
  Regs[3].Name := 'D3';
  Regs[4].Name := 'D4';
  Regs[5].Name := 'D5';
  Regs[6].Name := 'D6';
  Regs[7].Name := 'D7';
  Regs[8].Name := 'A0';
  Regs[9].Name := 'A1';
  Regs[10].Name := 'A2';
  Regs[11].Name := 'A3';
  Regs[12].Name := 'A4';
  Regs[13].Name := 'A5';
  Regs[14].Name := 'A6';
END InitRegNames;
(* ***** *)
BEGIN
  InitRegNames;
  SpyWindow := NIL;
END Snoop.

```

```

MODULE Spy;

(* ***** *)
(* Spy - A Task Control Block Snooper *)
(* *)
(* Written by Steve Faiwizewski, June 1988 *)
(* *)
(* Not to be used for commercial purpose *)
(* ***** *)

FROM Termination IMPORT ExitGracefully, AddTerminator;
FROM Snoop IMPORT WINDOWLEFT, WINDOWTOP, WIDTH,
  HEIGHT, SpyWindow, Observe;
FROM Nodes IMPORT Node, NodePtr, NTProcess;
FROM Heap IMPORT ALLOCATE, FreeHeap;
FROM TermInOut IMPORT WriteLn, WriteString, WriteCard,
  Write, WriteHex;
FROM Strings IMPORT StringLength;
FROM Tasks IMPORT Task, TaskPtr, CurrentTask,
  TaskState, FindTask, SignalSet, Wait;
FROM Interrupts IMPORT Forbid, Permit;
FROM Rasters IMPORT Jaml, Jam2, RastPortPtr;
FROM System IMPORT argc, argv, ExecBase;
FROM ExecBase IMPORT ExecBasePtr;
FROM Ports IMPORT MsgPortPtr, MessagePtr, GetMsg,
  ReplyMsg, WaitPort;
FROM Text IMPORT Text;
FROM Drawing IMPORT Move, Draw, SetAPen, SetBPen,
  WritePixel, RectFill, SetDrMd;
FROM Intuition IMPORT WindowFlags, WindowFlagsSet,
  IDCMPFlagsSet, IDCMPFlags,
  GadgetActivation,
  WindowPtr, CloseWindow, RemoveGadget,
  GadgetPtr, PropInfoPtr, ModifyIDCMP,
  IntuiMessagePtr, DoubleClick;
FROM SYSTEM IMPORT ADDRESS, ADR, WORD, LONGWORD, BYTE,
  TSIZE;
FROM IntuiCommon IMPORT OpenSimpleWindow;
FROM Conversions IMPORT ConvStringToNumber;
FROM AmigaDOSProcess IMPORT ProcessPtr;
FROM AmigaDOSExt IMPORT CommandLineInterfacePtr;
FROM InputEvents IMPORT IECodeLButton;
FROM SimpleGadgets IMPORT BeginGadgetList, EndGadgetList,
  LastGadget, AddGadgetProp,
  FreeGadgetList;

CONST
  PROPLEFT = 280;
  PROPTOP = 10;
  PROPWIDTH = WIDTH - PROPLEFT;
  PROPHEIGHT = HEIGHT - PROPTOP - 1;
  LetterHeight = 9;
  LetterWidth = 8;
  MaxNameLength = (PROPLEFT DIV LetterWidth) - 1;
  MaxDisplayLines = PROPHEIGHT DIV LetterHeight;

TYPE
  MyNodePtr = POINTER TO MyNode;

  MyNode = RECORD
    address : ADDRESS;
    next : MyNodePtr;
  END;

VAR
  TargetTask : TaskPtr;
  ExecBaseP : ExecBasePtr;
  MyGadList : GadgetPtr;
  Piptr : PropInfoPtr;
  Divisor : CARDINAL;
  PreviousSelectedLine : CARDINAL;
  PreviousSelectedItemPtr : MyNodePtr;
  Blanks : ARRAY[0..MaxNameLength-1] OF CHAR;
  CloseTheWindow : BOOLEAN;

(* ***** *)
PROCEDURE CopyList(n : NodePtr; VAR tail : MyNodePtr;
  VAR count : CARDINAL): MyNodePtr;
(* make a copy of the list while multitasking is FORBIDDEN *)
VAR
  tmp,
  head : MyNodePtr;
BEGIN
  head := NIL;
  tail := NIL;
  WHILE (n <> NIL) AND (n^.lnSucc <> NIL) DO
    INC(count);
    ALLOCATE(tmp, TSIZE(MyNode));
    WITH tmp^ DO
      address := n;
    END;
  END;

```



```

        next := head;
    END;
    head := tmp;
    IF tail = NIL THEN tail := tmp END;
    n := n^.lnSucc;
END; (* while n <> NIL *)
RETURN head
END CopyList;

(* ***** *)
PROCEDURE BuildTaskList(VAR total : CARDINAL) : MyNodePtr;
(* Build a list of all the tasks on the system *)
VAR
    tail,
    tail2,
    tmp,
    MyTaskList : MyNodePtr;
BEGIN
    Forbid;
    total := 0;
    MyTaskList := NIL;
    WITH ExecBaseP^ DO
    (* First get all the "ready" tasks *)
        MyTaskList := CopyList(TaskReady.lhHead,tail,total);
    (* Now get all the "waiting" tasks *)
        tmp := CopyList(TaskWait.lhHead,tail2,total);
        IF MyTaskList = NIL THEN
            MyTaskList := tmp
        ELSE
            tail^.next := tmp
        END
    END;
    Permit;
    RETURN MyTaskList;
END BuildTaskList;

(* ***** *)
PROCEDURE VerifyTaskIsReal(TargetTask : ADDRESS) : BOOLEAN;

(* Make sure that the task we'll be trying to spy on is *)
(* a real one (i.e. it's not a bogus address and the task *)
(* hasn't disappeared on us. *)

VAR
    t : MyNodePtr;
    found : BOOLEAN;
    total : CARDINAL;
BEGIN
    t := BuildTaskList(total);
    found := FALSE;
    WHILE (t <> NIL) AND NOT found DO
        found := t^.address = TargetTask;
        t := t^.next;
    END;
    FreeHeap;
    RETURN found;
END VerifyTaskIsReal;

(* ***** *)
PROCEDURE Min(x,y : CARDINAL) : CARDINAL;
BEGIN
    IF x > y THEN RETURN y ELSE RETURN x END
END Min;

(* ***** *)
PROCEDURE Len(s : ADDRESS) : CARDINAL;
(* Calculate the length of a string pointed to by s *)
VAR cp : POINTER TO CHAR;
    i : CARDINAL;
BEGIN
    cp := s;
    i := 0;
    WHILE cp^ <> 0C DO
        INC(i);
        cp := ADDRESS(LONGCARD(cp) + 1D);
    END;
    RETURN i;
END Len;

(* ***** *)
PROCEDURE PrintTaskName(RP : RastPortPtr; t : MyNodePtr;
    line, APen, BPen : CARDINAL);
(* Print a task's name. If it also happens to be a process *)
(* then print the process (command) name instead. *)
VAR
    tp : TaskPtr;
    pp : ProcessPtr;
    CliPtr : CommandLineInterfacePtr;
    NameP : POINTER TO CHAR;
    y, len : CARDINAL;
BEGIN
    tp := t^.address;
    NameP := tp^.tcNode.lnName;
    IF CHAR(tp^.tcNode.lnType) = CHAR(NTProcess) THEN

```

# Do you have a good idea for an article?

We can guide you  
from **idea** to **published article**

Please Contact:

**Editor**  
*Amazing Computing*  
 P.O. Box 869  
 Fall River, MA 02722  
 (508) 678-4200

```

pp := ProcessPtr(tp);
IF pp^.prCLI <> NIL THEN
    CliPtr := ADDRESS(LONGCARD(pp^.prCLI)*4D);
    NameP :=
        ADDRESS(LONGCARD(CliPtr^.cliCommandName)*4D);
    IF NameP^ = 0C THEN
        NameP := ADR('(No Command)')
    ELSE
        NameP := ADDRESS(LONGCARD(NameP) + 1D)
    END;
END
END;
y := 10 + line * LetterHeight;
SetAPen(RP^,0); SetBPen(RP^,0);
RectFill(RP^,5,y,PROPLEFT-2,y+LetterHeight);
SetAPen(RP^,APen);
SetBPen(RP^,BPen);
Move(RP^,5,y + LetterHeight - 2);
len := Len(NameP);
IF len > MaxNameLength THEN len := MaxNameLength END;
Text(RP^,NameP,len);
Text(RP^,ADR(Blanks),MaxNameLength - len);
END PrintTaskName;

(* ***** *)
PROCEDURE CleanUp;
VAR i : INTEGER;
BEGIN
    IF CloseTheWindow AND (SpyWindow <> NIL) THEN
        CloseWindow(SpyWindow^);
        SpyWindow := NIL
    END;
    IF SpyWindow <> NIL THEN
        i := RemoveGadget(SpyWindow^,MyGadList^);
    END;
    IF MyGadList <> NIL THEN
        FreeGadgetList(MyGadList^);
        MyGadList := NIL
    END;
    FreeHeap;
END CleanUp;

```

(continued)



```

(* ***** *)
PROCEDURE CalculateTaskFromItem(item : CARDINAL;
                                TaskList : MyNodePtr): ADDRESS;
(* Find out which task corresponds to position number 'item' *)
VAR
    t : MyNodePtr;
    i : CARDINAL;
BEGIN
    t := TaskList;
    FOR i := 1 TO item-1 DO
        t := t^.next
    END;
    RETURN t^.address
END CalculateTaskFromItem;

(* ***** *)
PROCEDURE SelectItem(item, FirstItem : CARDINAL;
                    RP : RastPortPtr; TaskList : MyNodePtr);
(* Highlight the name of the task the user just clicked on *)
VAR
    i,
    line : CARDINAL;
    t : MyNodePtr;
BEGIN
    IF PreviousSelectedItemPtr <> NIL THEN
        PrintTaskName(RP, PreviousSelectedItemPtr,
                      PreviousSelectedLine, 1, 0);
    END;
    line := item - FirstItem;
    t := TaskList;
    FOR i := 1 TO item-1 DO
        t := t^.next
    END;
    PrintTaskName(RP, t, line, 0, 1);
    PreviousSelectedItemPtr := t;
    PreviousSelectedLine := line;
END SelectItem;

(* ***** *)
PROCEDURE CalculateFirstItem(TotalTasks : CARDINAL) : CARDINAL;
(* Calculate which task is the first on the display *)
VAR FirstItem : CARDINAL;
BEGIN
    FirstItem := PIPtr^.VertPot DIV Divisor + 1;
    IF FirstItem > (TotalTasks + 1 - MaxDisplayLines) THEN
        FirstItem := TotalTasks + 1 - MaxDisplayLines
    END;
    RETURN FirstItem
END CalculateFirstItem;

(* ***** *)
PROCEDURE OpenTaskWindow(VAR Divisor : CARDINAL;
                        VAR MyProp : GadgetPtr;
                        VAR PIPtr : PropInfoPtr): SignalSet;
VAR
    i : CARDINAL;
BEGIN
    FOR i := 0 TO MaxNameLength - 1 DO Blanks[i] := ' ' END;
    BeginGadgetList;
    AddGadgetProp(PROPLEFT, PROPTOP, PROPWIDTH, PROPHEIGHT,
                  FALSE, TRUE, 1, 1, Divisor);
    MyProp := LastGadget;
    (* Add GadgImmediate so we get GadgetDown event *)
    INCL(MyProp^.Activation, GadgImmediate);
    PIPtr := MyProp^.SpecialInfo;
    MyGadList := EndGadgetList();
    SpyWindow := OpenSimpleWindow(WIDTH, HEIGHT, WINDOWLEFT,
                                  WINDOWTOP,
                                  ADR('Snoop: List of Tasks'),
                                  WindowFlagsSet(Activate, WindowDrag,
                                                  WindowDepth, WindowClose, NoCareRefresh),
                                  IDCMPFlagsSet(MouseButtons, GadgetDown,
                                                  GadgetUp, CloseWindow),
                                  MyGadList, NIL);
    SetAPen(SpyWindow^.RPort, 1);
    SetDrMd(SpyWindow^.RPort, Jam2);
    RETURN SignalSet(CARDINAL(SpyWindow^.UserPort^.mpSigBit));
END OpenTaskWindow;

(* ***** *)
PROCEDURE GetTaskFromUser(VAR task : ADDRESS) : BOOLEAN;
(* Display the list of tasks that are currently in the *)
(* system. *)
(* Wait for the user to either choose one task, or to *)
(* exit. *)
VAR
    sig,
    MySig : SignalSet;
    msg : IntuiMessagePtr;
    PreviousSecs,
    PreviousMicros : LONGCARD;
    MyProp : GadgetPtr;
    good,
    done : BOOLEAN;
    TotalTasks,
    FirstItem,

```

```

    PreviousItem : CARDINAL;
    TaskList : MyNodePtr;

(* ***** *)
PROCEDURE NeedUpdate() : BOOLEAN;
(* Check if display needs to be refreshed *)
VAR NewFirstItem : CARDINAL;
BEGIN
    IF TotalTasks <= MaxDisplayLines THEN RETURN FALSE END;
    NewFirstItem := CalculateFirstItem(TotalTasks);
    IF NewFirstItem = FirstItem THEN
        RETURN FALSE
    ELSE
        RETURN TRUE
    END
END NeedUpdate;

(* ***** *)
PROCEDURE DisplayIt(RP : RastPortPtr);
(* Display the list of tasks *)
VAR
    t : MyNodePtr;
    i,
    LastItem : CARDINAL;
BEGIN
    PreviousSelectedItemPtr := NIL;
    IF TotalTasks <= MaxDisplayLines THEN
        FirstItem := 1
    ELSE
        FirstItem := CalculateFirstItem(TotalTasks)
    END;
    LastItem := Min(FirstItem + MaxDisplayLines - 1,
                  TotalTasks);
    t := TaskList;
    FOR i := 1 TO FirstItem-1 DO
        IF t <> NIL THEN t := t^.next END
    END;
    FOR i := FirstItem TO LastItem DO
        IF t = NIL THEN RETURN END;
        PrintTaskName(RP, t, (i - FirstItem), 1, 0);
        t := t^.next
    END;
END DisplayIt;

(* ***** *)
PROCEDURE CalcItem(x, y : INTEGER) : CARDINAL;
(* Find out which task was selected. *)
(* Return the task's position number in the list of tasks. *)
VAR item : CARDINAL;
BEGIN
    item := CARDINAL(y + 1 - LetterHeight) DIV LetterHeight;
    IF item > (MaxDisplayLines - 1) THEN
        item := MaxDisplayLines - 1
    END;
    IF item <= TotalTasks THEN
        RETURN item
    ELSE
        RETURN 0
    END
END CalcItem;

(* ***** *)
PROCEDURE ProcessIntuiMsgs(msg : IntuiMessagePtr;
                          VAR done, good : BOOLEAN);
VAR
    item : CARDINAL;
    secs,
    micros : LONGCARD;
    address : ADDRESS;
    class : IDCMPFlagsSet;
    code : CARDINAL;
    mx, my : INTEGER;
BEGIN
    WITH msg^ DO
        class := Class;
        address := IAddress;
        code := Code;
        mx := MouseX;
        my := MouseY;
        secs := Seconds;
        micros := Micros;
        ReplyMsg(msg)
    END; (* with *)
    IF CloseWindow IN class THEN (* User wants out *)
        done := TRUE
    ELSIF GadgetDown IN class THEN
        IF (address = MyProp) AND
            (TotalTasks > MaxDisplayLines) THEN
            (* User clicked on slider, so start listening to IntuiTicks *)
            ModifyIDCMP(SpyWindow^, SpyWindow^.IDCMPFlags +
                       IDCMPFlagsSet(IntuiTicks))
        END
    ELSIF GadgetUp IN class THEN
        IF address = MyProp THEN
            (* User released slider, so stop listening to IntuiTicks *)
            ModifyIDCMP(SpyWindow^, SpyWindow^.IDCMPFlags -
                       IDCMPFlagsSet(IntuiTicks))
        END
    END
END

```



```

ELSIF IntuiTicks IN class THEN
(* Got a clock tick, so check if we need to refresh display *)
IF NeedUpdate() THEN
DisplayIt(SpyWindow^.RPort)
END
ELSIF MouseButton IN class THEN
IF code = IECodeLButton THEN
item := CalcItem(mx,my) + 1;
item := FirstItem + item - 1;
IF (PreviousItem = item) AND
DoubleClick(PreviousSecs,PreviousMicros,
secs,micros) THEN
(* User picked a task to spy on *)
task := CalculateTaskFromItem(item,TaskList);
done := TRUE;
good := TRUE
ELSE
(* User is thinking about spying on a task, *)
(* so let's highlight it *)
PreviousItem := item;
PreviousSecs := secs;
PreviousMicros := micros;
SelectItem(item,FirstItem,SpyWindow^.RPort,
TaskList)
END (* if PreviousItem ... *)
END (* if code = IECodeLButton *)
END;
END ProcessIntuiMsgs;
(* ----- *)
BEGIN (* GetTaskFromUser *)
PreviousSelectedItemPtr := NIL;
good := FALSE;
done := FALSE;
TaskList := BuildTaskList(TotalTasks);
IF TotalTasks <= MaxDisplayLines THEN
Divisor := 0FFFFH
ELSE
Divisor := 0FFFFH DIV (1+ TotalTasks - MaxDisplayLines)
END;
MySig := OpenTaskWindow(Divisor,MyProp,PIptr);
DisplayIt(SpyWindow^.RPort);
REPEAT
sig := Wait(MySig);
msg := GetMsg(SpyWindow^.UserPort^);
WHILE (msg <> NIL) DO
ProcessIntuiMsgs(msg,done,good);
msg := GetMsg(SpyWindow^.UserPort^);
END; (* while *)
UNTIL done;
CloseTheWindow := FALSE;
CleanUp;
CloseTheWindow := TRUE;
RETURN good
END GetTaskFromUser;
(* ***** *)
PROCEDURE Main;
VAR
good : BOOLEAN;
Myself : TaskPtr;
BEGIN
Myself := FindTask(CurrentTask);
IF argc < 2 THEN
good := GetTaskFromUser(TargetTask);
ELSIF argc > 2 THEN
WriteString('Format: ');
WriteString(argv[0]^);
WriteString(
' xxxx\nwhere xxxx is the hex address of a task\n');
good := FALSE
ELSE
good := ConvStringToNumber(argv[1]^, TargetTask,
FALSE, 16);
IF NOT good THEN
WriteString('Invalid data in address field!\n');
ELSIF LONGCARD(TargetTask) MOD 4D <> 0D THEN
WriteString('Invalid address!\n');
good := FALSE;
ELSIF TargetTask = Myself THEN
WriteString('Can't snoop on myself!!\n');
good := FALSE;
END;
END;
IF good THEN good := VerifyTaskIsReal(TargetTask) END;
IF good THEN
Observe(TargetTask)
END;
END Main;
END Spy.
(* ***** *)
BEGIN
ExecBaseP := ExecBase;
CloseTheWindow := TRUE;
SpyWindow := NIL;
MyGadList := NIL;
AddTerminator(CleanUp);
Main;
ExitGracefully(0)
END Spy.

```



## Time saving utilities for EVERY Amiga owner!

- Create multiple preference settings.
- Quickly and easily restore them as needed.
- Edit, Sort, Delete and Undelete any of them.
- Fast, Fun and Easy to use, from WorkBench or CLI.
- Runs on ALL Amigas . . . 500, 1000 and 2000 (V1.2 required).
- Multi-tasking.
- Not copy protected.
- Plus . . . 3 other time saving utilities are included.

**Available NOW!!!**

Just send \$29.95 plus \$2.00 S&H to:

**SOFT·LINK** inc. P.O. Box 304, Coventry, RI 02816

Dealer inquiries invited.

Amiga is a registered trademark of Commodore-Amiga, Inc.

## STUFF-IT!

Design your own menus and use them with your favorite Amiga software!

- Each menu is like having thousands of user definable, descriptively labeled, function keys
- Quickly and easily create all the menus you like
- Modify any of your menus, as often as needed
- Each menu can restore other Stuff-It menus
- Also included are a few menus we've created to get you started (CLI-Menus, ED-Menus, etc.)
- Runs on ALL Amigas...From WorkBench or CLI (V1.2 or higher required)
- No hardware requirements • NOT copy protected

**Stuff-It is a MUST for ALL Amiga owners, beginner to expert!**

Just send \$39.95 plus \$2.00 shipping & handling to:

**SOFT·LINK** inc.

P.O. Box 304 Coventry, RI 02816

Amiga is a registered trademark of Commodore-Amiga, Inc.



### Inside your Genlock

"Any sufficiently advanced technology is indistinguishable from magic." Arthur C. Clarke

Does it bug you to use a device and not have any idea of what makes it tick? It bugs me too, so this month we're going to crack open that black box called a "genlock" and see how it works.

First, just what does the term "genlock" mean? Where did it come from? It's a broadcast television term which means "to synchronize to an external signal." This includes synchronization of the vertical, horizontal and subcarrier timing signals. It's impossible to "mix" or combine in any way two video signals unless they are synchronized. Any fades, wipes, overlays etc. are out of the question unless we can somehow sync the two. Until digital

technology came around it took a lot of hardware to manage the task. Now it's much simpler.

### Genlocking through the past

While most current video gear—except VCRs—has genlocking capability, computers rarely do. One of the first computers with a genlocking capability was the Mindset. An IBM kinda-compatible machine, it featured special chips to handle the graphics for an enhanced array of colors and resolution. Yes, it was very much like an Amiga but much more expensive while not as capable.

Another such computer was the Sony SMC-70, which was intended for use as a desktop video machine long before the category even existed. It too fell short of the mark.

Released before both of those was the VIC-20, the only other genlockable computer I know of. This wasn't well known, but feeding sync signals to the correct pin of the video connector would cause it to "lock-up" or synchronize to an external source.

### Genlocking the Amiga

Since genlock requires that the sync signals of each source be "in sync" we must examine the source of the Amiga's sync for its video. These signals are based on the Amiga's master or "system" clock.

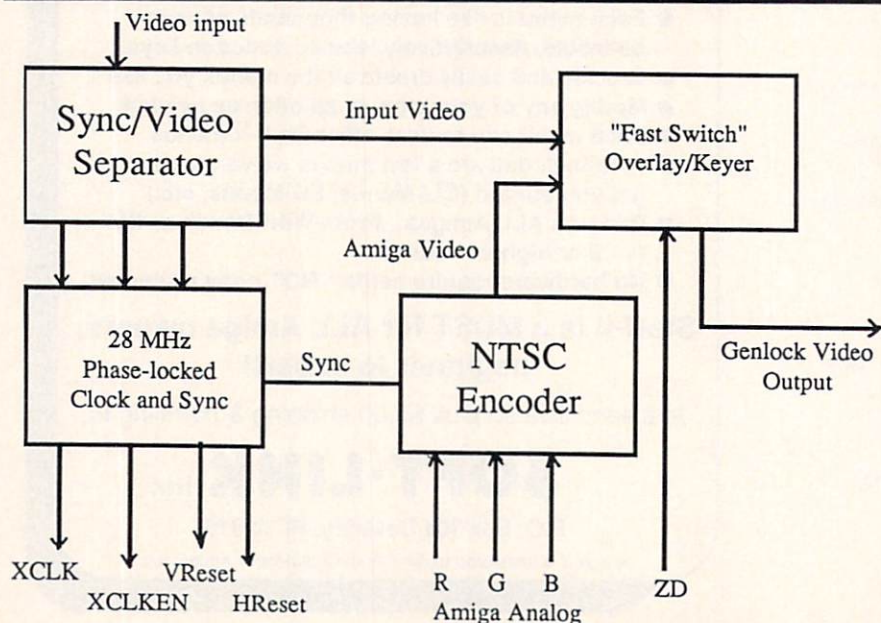
The master clock of the Amiga runs at 28.8 MHz, which is divided down to 14.4 MHz and 7.17MHz as needed for certain chips. All these frequencies are even multiples of the highest frequency sync signal used for color video, the color subcarrier at 3.58MHz. The Paula and Denise chips use these frequencies to produce the other necessary synchronization signals (vertical and horizontal), so it becomes necessary to control the system's master clock in order to control its video sync signals.

Via the RGB port the Amiga allows us to feed the Amiga another master clock source. This pin is known as the XCLK or external clock. Another pin, XCLKEN, allows us to tell the Amiga to use the external clock. All genlocks provide this clock.

Let's now look at the block diagram of the genlock. Please bear in mind that this diagram is VERY simplified and quite a bit more happens than what's shown.

Inside the genlock is the external clock we mentioned above. It runs at 28.8Mhz, same as the Amiga's master clock. The one difference is that it can be phase-locked or "synchronized" to another signal. This is the first step to becoming genlocked.

Figure One— Block Diagram of a Genlock





# SYNC TIPS

## INSIDE YOUR GENLOCK

BY ORAN SANDS

The input video signal (whether a camera or video player) is separated into its components: the video information, the horizontal sync, the vertical sync and the color subcarrier. These signals are then used to "drive" the external clock. Now the Amiga and the input video signal are running at the same rate, and if the input video signal's subcarrier varies or drifts, so does the Amiga's, thus maintaining the synchronization.

The Amiga however isn't necessarily timing the vertical and horizontal intervals of its picture the same as the input video signal (which I'll just call the input since I'm tired of typing the whole thing). To correct this the genlock must generate correctly timed horizontal and vertical sync from the input. These reset signals as they're referred to are applied to the horizontal and vertical pins on the RGB port.

Usually these pins are outputs and send these signals to your monitor. But when a signal is fed to them, they sense it and become inputs instead. The Paula chip now uses these signals to control the timing of the vertical and horizontal timing of the Amiga's picture.

With these signals now timed correctly, the Amiga's video is now "genlocked" with the input. Now we can try to combine them in some manner.

### Keying

The process of "keying" is usually what we call genlocking. It is the process whereby we combine or overlay the Amiga's graphics with the input signal. In reality, we could do much more, but for now let's stick with the overlay function.

When keying two graphics together, we need some basis for determining which picture is to be displayed when. On most of the available genlocks, the determination is

based on the color # of the current pixel, or in some cases its luminance value.

The main function of the overlay portion of the genlock is that of a two-way switch (again a simplification). If the switch is thrown one way we "see" the input video, if it is thrown the other way we see the Amiga signal. This switching can occur as often as every 70 nanoseconds, hence the name "fast switch".

Since a video picture isn't displayed all at once but a little at a time, with one piece following another on a horizontal line, we need to continually decide which way to throw that switch. As we look along each horizontal line of video we encounter each individual pixel. If that pixel is represented by color #0 then we decide to display the input signal. If it's a color other than color #0 i.e. colors #1-31, then we show the Amiga picture.

The Amiga creates a signal to represent this: It turns on to say "show the input signal" and off to say "now show the Amiga's picture". This on/off signal is called ZD (or Zed D) and is found on the RGB port. The genlock uses this signal to control its overlay function.

That's a simplification of your genlock's function, but believe me, a great deal more goes on to make sure that all the signals are correctly timed.

### Options

There are several options available to genlock manufacturers as to how to treat the video signal. Some of these bear mentioning.

The externally applied video is always in a composite video form, hopefully meeting RS-170A specifications for timing. This signal can be broken down into its original components, the red, green and blue information (as well

as the timing info). These signals can be "fast switched" the same as the composite video signals so that the overlay may be performed with both the input signal and the Amiga signal in RGB form.

This can be an advantage to the user. If the input signal can be turned into RGB signals then it may be viewed on a RGB monitor. If the final overlaid video is also in RGB form, it too may be viewed on the RGB monitor. The average genlock user is usually a home user with only a TV receiver and his Amiga monitor to view his creations. This way the Amiga monitor can be used in a double role, computer and video monitor. The original Amiga 1300 genlock had a switch to allow you to switch between the Amiga picture, the input video or the overlay combination.

There is a disadvantage to this approach however. The input video signal must be decoded to obtain the RGB information. Unless this is done utilizing "comb" filtration or other such procedures, the resolution of the input signal, regardless of how high, is reduced to no better than about 220 lines. It's a problem with decoding processes and occurs in monitors, TV's or any device that must decode the composite video signal.

For professional work this won't do. It doesn't make any sense to destroy the signal you've worked so hard to produce. There are many other ways that the signal's resolution will suffer so it behooves you to keep it as high as possible at all times. (Look for a column on resolution concerns next month).

Fortunately very few genlocks—and none of the "professional" models—use this approach. For the home use, you may never notice the problem, so

(continued)



don't sweat it. Professionals should stick to genlocks that pass through the composite signal untouched.

### **Genlock Myth-conceptions**

I keep hearing untrue statements about genlocks. These statements seem to be repeated by everyone like a chant. Let's dispel a few.

*"You need an input signal to get an Amiga picture out of a genlock."*

I hear this one more than any other. I've tested just about every genlock sold and I haven't found one yet that needed an external signal applied to it in order to make it work. It IS possible to make one that way but it would be a rather foolish thing to do. If there is any basis to this whatsoever it's because some genlocks that were out of adjustment were shipped by several different manufacturers. The external master clock wouldn't start running until "nudged" by the external signal. Once running, they would continue to operate with or without the external signal applied. Such units should be returned to the manufacturers and repaired. Just to be on the safe side, check the literature carefully.

*"You need a genlock to get video out of an Amiga."*

Well, it's certainly one way to get composite video out of an Amiga 500 or 2000 but hardly the ONLY answer. If you don't need the genlocking/overlaying ability of a genlock then check into the many video adaptors. I can recommend the CMI VI-500 or 2000 for more critical work than home movies.

*"A genlock will let you get video from a VCR into your Amiga."*

Your video signal never gets any closer to your Amiga than 2 inches. The genlock does NOT allow the Amiga to play with or otherwise affect your external video signal. In fact, the reverse is true—the external video signal affects your Amiga by changing its timing to match.

### **Cautions**

Just a few words of warning about using genlocks.

You can use the original Amiga 1300 genlock on both the 1000 and 2000 computers, but not on the 500. The 1300 can be used on the 2000 by merely propping up the computer about an inch and then sliding the 1300 under it. You could do that on the 500 but DON'T! The voltages that appear on the RGB port of the 500 are not the same as the 1000 or 2000. I don't know why, but they are. Plugging in the 1300 could damage the 500, your genlock or both.

Speaking of power supplies, there is a newly appearing problem with the RGB port voltages on the 2000. It seems that the +5 volt supply on that port is separate from the 5 volt system supply for the rest of the computer. This separate power supply is called the "user +5 volts supply" and also supplies voltages to several other ports as well. It is very possible to overload this supply, particularly if you have a genlock and other devices such as a modem hooked up. If this power supply burns out, the system 5 volt supply will continue to operate. The machine will work but your genlock will appear not to. If testing the genlock on another machine finds that it's ok, then I'd check the 5 volt pin on the RGB port. The power supply is usually heavy enough to carry several devices but some genlocks pull a lot of power.

### **Unstable Signals**

Since a genlock locks to the externally applied video signal, the stability of that signal will determine the stability of your combined Amiga/input video signal. This means you should avoid unstable external sources of video, which may include tuners and VCR's. The normal output of a tuner is quite stable unless you're having trouble receiving the station. If there's a good deal of snow in the picture, or if the picture fades in and out, then you may have problems. As for VCR's, your usual home VCR will function just fine unless... you're using a tape that is a copy of a copy of a...well, you get the idea.

The output of any VCR is somewhat unstable. This instability is multiplied by copying the tape to

another VCR. Each generation down from the original tape will include every timebase error from previous generations as well as its own. This is the reason for using timebase correctors, devices that realign the timing components of the signal so they are stable again. Timebase correctors are the way to go if you can afford them but are not really necessary unless you're doing professional work. Just don't use tapes that are copies of copies of, well, you know.

Tapes that are not "solid" video from end to end can pose a problem. If a blank spot on the tape, snow, or a damaged spot runs by while you're using your genlock, expect some picture disturbance. This is roughly equivalent to removing the input signal and then reapplying it. The external clock of the genlock must switch to free-run mode and then relock to the newly applied input video. This can be quite a drastic change and will make for big timebase errors.

I hope I've made genlock operation a little clearer for you. In reality, the genlock is a very complex device. There are many special relationships between the various aspects of the video signal that need to be maintained to assure good video. These relationships are spelled out in the standard for color composite video signals, better known as RS-170A. Unfortunately, not all the available genlocks rigorously follow the standard. If you need high quality video, then buy one of the more expensive units. You really DO get what you pay for. Besides, let's put this in perspective: The "high" cost of genlocks is considered petty cash by the broadcast industry.

Remember, if you have any questions or comments, please send them to me in care of this magazine or via EMAIL on PeopleLink to ID OJSANDS. I can't guarantee a personal answer, but don't be surprised to see it in this column. That's it for this month. Next month (God willing) we'll look at resolution.

*PS. Look for a review of the Magni 4004 genlock system soon! These people have solved problems that others didn't know existed.*



# On the Crafting of Programs

## The Proposed ANSI C Standard

by David J. Hankins

In this month's column, I'll discuss the draft proposed ANSI C standard. Since both Manx and Lattice have pledged support of the standard, this column will probably be of interest to those who program in C on the Amiga.

### ANSI C

ANSI is an acronym which stands for "American National Standards Institute". In 1982, ANSI formed subcommittee X3J11 to develop a standard definition for the C programming language. X3J11's standard is still in a "draft proposed" state pending endorsement of its international counterpart, ISO. So, until ISO ratifies the standard, ANSI C is more properly referred to as "draft proposed ANSI C". For sake of brevity however, the terms "ANSI C" and "draft proposed ANSI C" will be used interchangeably throughout the remainder of this column.

### Why Do We Need a New Standard?

One reason for C's popularity is that it is a very portable language - that is, a C program written for one type of computer (let's say an IBM PC) will often run with little or no modification on another type (the Amiga, for example). This is possible because widespread agreement exists among compiler vendors on how the language should be implemented. Given this, why do we need a new standard for C?

### Standardized Interpretation

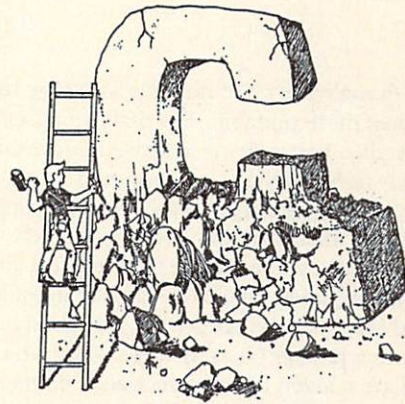
One reason that comes to mind is that, while there is agreement on the basic "definition" of C, disagreement exists on some of the finer details. For example, consider the statement

```
x = (y - 30000) + (z + 0) ;
```

where x, y, and z are ints, on a machine with 16-bit integers. Under the existing standard (as defined in Kernighan and Ritchie's "The C Programming Language"), compiler vendors are free, but are not required to implement the arithmetic sum as

```
x = (y + z) - 30000 ;
```

By combining the constants -30000 and 0 at compile time, execution speed is improved. Unfortunately, this sort of optimization may lead to problems. Consider, for example, that 30000 + 30000 is too large to be represented with only 16 bits (1 bit is reserved for the sign); overflow results. On certain machines (not the Amiga), overflow causes a trap to occur. When this happens, normal program flow is interrupted, and a trap routine is executed instead. Thus, assuming y and z were both equal to 30000, the expression



```
x = (y - 30000) + (z + 0) ;
```

would be evaluated without any problems on compilers that do not regroup constants, and x would be assigned the value 30000. But on compilers which regroup the expression as we saw earlier, i.e.

```
x = (y + z) - 30000 ;
```

overflow would occur when attempting to evaluate y + z and the program would terminate.

ANSI C solves this problem by forbidding the regrouping of expressions in cases such as this on machines that trap on overflow.

### Error Checking

Another, and perhaps more compelling, reason for a new standard is that C is too lax in its error checking. For example, given a function f1() which takes three short integer arguments,

```
f1( arg1, arg2, arg3 )
short arg1, arg2, arg3 ;
{
    .
    .
    .
}
```

C allows you to call f1() with the wrong number of arguments

```
short a, b ;
y = f1( a, b ) ;
```

or worse yet, to call f1() with the wrong argument types

```
short a, b ;
y = f1( a, b, 32 ) ;
```

In either of these instances, C will issue no error or warning messages of any sort, leaving these inconsistencies to be resolved at run-time. Fixing errors at run-time is invariably more difficult than at compile-time. With run-time errors, a programmer must first determine where in the code the error exists. Contrast this with compile-time errors. Compile-time errors are accompanied by nice warning/error messages which pinpoint mistakes with no effort on the programmer's part.

To overcome this liability, the new C standard introduces function prototypes. Prototypes are discussed in more detail later in this column.

(continued)



## Portability

Finally, even though C is quite portable as it now stands, it could be made even more portable. The draft proposed ANSI C standard does just this. Portability is increased under the standard by precisely defining the C language and its run-time libraries, and by extending the language to overcome various pitfalls. One "extension" that ANSI C introduces is a new variable type, `size_t`, which will either be typedef'd as an unsigned int or as an unsigned long in the header file `stddef.h`. However, unsigned long must be used when unsigned int is not large enough to store a pointer (as is the case with 16-bit ints on the Amiga - 32 bits are needed to accommodate a pointer value). To see how `size_t` increases portability, consider the following function call:

```
void *mem_ptr, *malloc() ;
unsigned long size ;
size = 80000 ;
mem_ptr = malloc( size ) ;
```

Using Lattice C, the call to `malloc()` works fine - ints are 32-bits in Lattice and 80000 can be represented as an int. However, with Manx C, problems arise. Manx defines the `malloc()` function as follows:

```
void *malloc( size )
int size ;
{
    .
    .
    .
}
```

Given this definition, the largest memory block we can allocate using Manx's 16-bit int default is  $(2^{16})-1$ , or 65,535 bytes. However, our call specifies 80000 bytes.

Under the ANSI C standard, Manx will have to redefine `malloc()` as follows:

```
typedef unsigned long size_t ;
void *malloc( size )
size_t size ;
{
    .
    .
    .
}
```

Given the ANSI C definition, the call to `malloc()` under Manx C would work. ANSI C increases portability, since both Lattice and Manx would accept the `malloc()` function call.

Having justified the need for a new C standard, the remainder of this column is devoted to some of the more interesting aspects of ANSI C.

## Function Prototypes

As we saw earlier, C is too permissive in its error checking. Errors which could easily be flagged at compile-time are let go, leaving the programmer to track them down at run-time, an often formidable task. Consider, for example, a call to the Amiga `OpenLibrary()` function to open the Intuition library:

```
#define INTUI_VERSION 29
struct IntuitionBase *IntuitionBase ;
void *OpenLibrary() ;
IntuitionBase = OpenLibrary( "intuition.library",
    INTUI_VERSION ) ;
```

With Lattice, the call to `OpenLibrary()` works fine, but, using Manx, the call brings an unwelcome visitation from the Guru. Why? The compiler compiled the code correctly, without any errors or warnings. Common sense (or is it emotional conviction?) tells us that no warnings and no errors mean that our code **MUST** be right, especially since it works on another vendor's compiler.

The problem in the above call is that `OpenLibrary()` expects two 32-bit arguments: a pointer to the name of the library to be opened, and the lowest acceptable version of the library. However, with Manx C, `INTUI_VERSION` is a 16-bit quantity. To get this code to work under Manx, the `#define` statement would have to be changed to:

```
#define INTUI_VERSION 29L
```

The "L" at the end of the number 29 causes the constant `INTUI_VERSION` to be a 32-bit long.

ANSI C introduces the concept of function prototypes to avoid problems like this. Function prototypes specify how many arguments must be passed to a function, as well as the types these arguments must possess. Using prototypes, errors such as the one mentioned above are caught at compile-time.

To see how prototypes work, let's rewrite the call to `OpenLibrary()` as follows:

```
#define INTUI_VERSION 29
struct IntuitionBase *IntuitionBase ;
void *OpenLibrary( char *, long ) ;
/* OpenLibrary() declaration
 * using prototypes
 */
```

Note the `OpenLibrary()` declaration statement. This statement is a function prototype which specifies that the first argument to `OpenLibrary()` must be of type pointer to char, and the second of type long. When we try to compile this code with an ANSI C compatible compiler, we will get an error since the second argument to `OpenLibrary` must be of type long, whereas `INTUI_VERSION` is of type int.

Function prototypes are also used in function definitions. Consider for example the factorial function (the factorial of 0 is defined to be 1, the factorial of 1 is defined to be 1, the factorial of 2 =  $2 * 1$ , the factorial of 3 =  $3 * 2 * 1$ , the factorial of 4 =  $4 * 3 * 2 * 1$ , etc.). Under the C standard described in "The C Programming Language", the factorial function might be written as:

```
unsigned int
factorial( x )
unsigned int x ;
{
    unsigned int factorial() ;

    if( x )
        x = x * factorial( x - 1 ) ;
    else
        x = 1 ;
    return( x ) ;
}
```



Using prototypes, factorial() could be rewritten as:

```
unsigned int
factorial( unsigned int x )    /* prototype in definition */
{
    unsigned int factorial( unsigned int ) ; /* prototype in
                                              * declaration
                                              */

    if( x )
        x = x * factorial( x - 1 ) ;
    else
        x = 1 ;
    return( x ) ;
}
```

### Function Prototypes Versus Lint

Lint is used to spot various kinds of anomalies in C programs. Lint examines C source code and reports suspicious statements (in which case warnings are issued) as well as incorrect statements (these provoke error messages). It seems to be a common misconception among beginning C programmers that function prototypes make lint obsolete. This is simply not true. Consider, for example, the expression

```
if( x = 3 )
    do_something() ;
```

In the above expression, it is quite likely that instead of the assignment operator "=", the equivalence-testing operator "==" was intended. Lint will faithfully issue a warning when it encounters expressions like this, while prototypes do nothing to prevent such errors.

In addition to prototypes, the ANSI C standard introduces several other new concepts, some of which are discussed below.

### Stringization

ANSI C allows programmers to substitute arguments within quotes through the use of the "stringization" operator, "#" (use of the stringization operator is limited to macro expressions). For example, given the macro

```
#define printit( number
    printf( #number " equals %d", number )
```

the following statement

```
printit( x ) ;
```

becomes

```
printf( "x equals %d", x ) ;
```

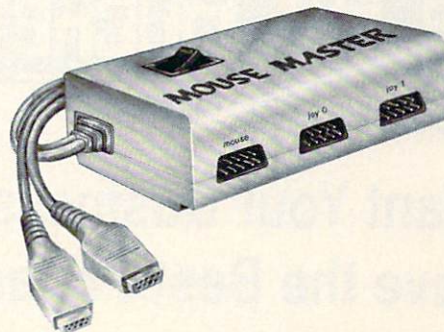
Use of "#" in a macro causes the formal argument name following "#" to be enclosed in quotes. Thus, #number becomes "x". Note that the above example also illustrates another new feature of ANSI C: adjacent character strings are merged. Thus,

```
printit( x ) ;
```

becomes

```
printf( "x" equals %d", x ) ;
```

## Now For The Amiga!



Are you tired of fumbling under or behind your computer to swap your mouse and joystick cables? Are your cable and computer connectors worn out from all the plugging and unplugging? Then Mouse Master is a must for you!

**\$39.95\***

**Practical Solutions®**

1930 E. Grant Rd.,  
Tucson, AZ 85719



\*Retail price does not  
include shipping & handling.

602-884-9612

**MOUSE MASTER**

after macro expansion, which then becomes

```
printf( "x equals %d", x ) ;
```

after string merging.

### Token Merging

Another feature new to C is token merging. The tokenizing operator "##" causes text on either side of the ## to be merged into a single token. For example, given the macro

```
#define Yindex( index )    y##index
```

the statement

```
Yindex( 1 ) = Yindex( 2 ) ;
```

becomes

```
y1 = y2 ;
```

(continued)



# The Bit Bucket

## COMPUTER STORE

**We Want Your Business!!  
We Have the Best Prices!!**

**Oldest Commodore Dealer in the Area!!**

**2 Locations to Serve You**

- \* Software
- \* Hardware
- \* Service
- \* Information



**1294 Washington Street  
W. Newton MA 02165  
617-964-3080**

**621 Boston Post Road  
Sudbury MA 01776  
508-443-9731**

**Authorized Commodore  
Amiga Dealer and Com-  
modore Service Center**

### Trigraphs

Perhaps of less interest to Amiga programmers are trigraphs. Trigraphs are used so that programs may be written using only the ISO 646-1983 Invariant Code Set characters (ISO 646-1983 lacks the following characters: "[", "{", "^", "#", "]", "}", "|", and "~"). To represent these characters, ANSI C introduces the following trigraphs:

Trigraph	Character Represented	Trigraph	Char Represented
??(	[	??)	]
??<	{	??>	}
??/	\	??!	
??'	^	??~	~
??#	#		

Thus, the statement

```
x[ 5 ] = 10 ;
```

could equivalently be written as

```
x??( 5 ??) = 10 ;
```

for systems which lack the "[" and "]" characters.

### Lattice, Manx, and ANSI C

If you're planning to use ANSI C right away, here's some good news. Although neither Lattice nor Manx currently support the draft proposed ANSI C standard, Lattice 4.0 does support function prototypes. Both Lattice and Manx promise to fully support ANSI C in the very near future. When I talked to Lattice in October, company representatives indicated that Version 5.0 of their compiler, due out in November 1988, would support all ANSI features except trigraphs. Not to be outdone, Manx expects to release Version 4.0 of their C compiler in December 1988. The Manx compiler should also support nearly all features of ANSI C.

### References

Readers wishing to learn more about ANSI C may be interested in reviewing the following books/periodicals:

Kernighan, Brian W. and Ritchie, Dennis M.; "The C Programming Language" 2nd ed.; Prentice Hall

Long regarded as the "C bible", this book has been revised in its second printing to conform to the ANSI C standard. Probably not a good book for beginning C programs, it is nevertheless a classic, capturing the beauty and elegance of C.

Harbison, Samuel P. and Steel, Guy L.; "C a Reference Manual", 2nd ed.; Prentice Hall

This book is an excellent all-around reference to the C language. Chapter 11 is devoted to ANSI C.

"The C Users Journal"; R&D Publications, Inc.

Columns by P. J. Plauger and Thomas Plum keep readers of this magazine well-informed of the latest developments in ANSI C.

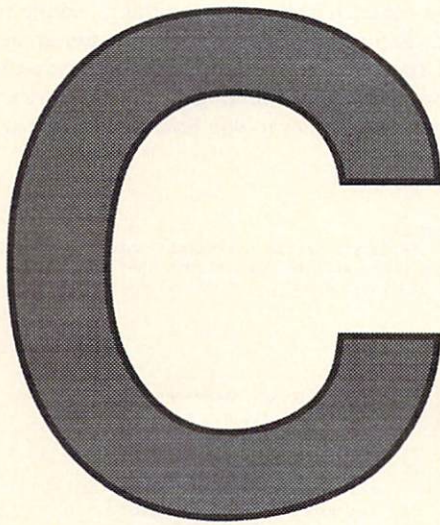
### Next Month

In next month's column I hope to take a look at some of the "tools" programmers use in creating programs. Among the tools examined will be diff, grep, make, splat, and touch. So until then . . .

```
printf( "Goodbye world!" ) ;
```

•AC•





# Notes

## *from the C Group*

### *Introduction to Unions*

*by Stephen Kemp*

The construct called "unions" can sometimes confuse newcomers to the C language. I don't think it is because unions are hard to use or define. I think it is because the concept behind unions can be hard to explain, which makes them hard to understand. The major difficulty involved in explanations is probably the lack of a good example that everyone can understand. This said, I will take my stab at explaining unions to the novice.

Most textbooks lump the discussion of unions in with their discussion of structures (another tough concept for newcomers). This is not because unions are part of structures or vice versa. Rather, it is probably because of the similarity in the way the two data types are defined and referenced. Additionally, it may be because using structure IN unions may be the best example of how to demonstrate the effectiveness of using unions (more on this later).

Unions in C are used to define "different" ways of referencing the "same" location. This is where the concept gets hard to explain, so let's use a "real life" example. Suppose you have a drinking glass that contains one of these refreshments: Orange juice, lemonade, tea, or cola. To make the simile with C, assume the glass represents a union's location. Although 4 refreshments (variables) have been identified, only one can occupy the location at any particular time. These 4 items are the "members" of the union. Now you are probably saying, "Yea, but I know a glass of orange juice from a glass of cola." Let's hope so, because you will have to do the same in your programs. Suppose you offer someone a glass of cola and when they begin to drink it they say, "Hey, wait a minute! This is orange juice not cola!" That is more warning than you might get from your program, if you don't know which variable type occupies the space at a given time.

The union definition looks almost exactly as that of a structure. The word "union" identifies what you are about to define. You can either use a union tag (which allows you to define other variables to be of the same union type) or just

define a union variable directly by naming it. Using tags is probably most common, so that is what I will use in the examples. A pseudo-example definition might look like this:

```
union tag_name {  
    type variable;  
    type variable;  
    type variable;  
};
```

As you can see, the definition is almost identical to that used for structures. The elements of the array are referenced similar to structures too. The variable name is identified first and then the member name. The two names are separated by the period or in the case that a pointer to a union is being used the symbol, ->, is used.

Actually, I have probably pressed the subject that only one element occupies the variable location at a given time a little too much. There are legitimate reasons why you might use a variety of ways to reference the same location. As an example, let's suppose you want to be able to individually address the bytes contained in a long variable. The definition would look something like this example.

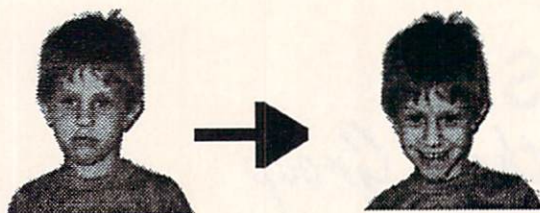
```
union DUO {                /* union tag and definition */  
    long lnum;              /* long variable */  
    char bytes[sizeof(long)]; /* byte array */  
}                            /* close of union definition */
```

With the union tag, DUO, we can define any number of variables that can be referenced as a long number or as an array that contains the number of characters found in a long. (We know that 4 characters make a long in this environment, but using the size of operator is still preferred.) If you wanted to know what number would be made if all the bytes in a long where the same, we could find out.

*(continued)*



# EDUSOFT™



## Enlighten Your Child Introducing..... Crayons And Rainbows

The Amazing Talking Computer teaches your preschooler about colors as your child plays a game to help the crayons paint a rainbow. Animated characters offer many challenges in color recognition and identification by providing entertaining scenes and situations.

Order our preschooler software from us or from a dealer near you.

**The Trumor Company**  
299 Coon Rapids Blvd NW  
Suite 103  
Coon Rapids, MN 55433  
(612)785-2289

An all Amiga company

For All Amigas. 512K memory required. More recommended.

AMIGA is a trademark of Commodore - AMIGA, Inc.

```
/* This function uses the DUO union definition to set the bytes */  
/* of a long to the value passed and then prints the long */
```

```
set_long(value)  
char value;  
{  
    short cnt;  
    DUO my_var;  
  
    for(cnt = 0; cnt < sizeof(long); cnt++)  
        my_var.bytes[cnt] = value;  
  
    printf("The bytes set to %d causes the long value to be %ld\n",  
        value, my_var.lnum);  
}
```

Although this function demonstrates how to use the union, it is probably not the most common use. As I have already mentioned, unions are usually used with structures. The reason behind this is because it usually takes some type of "identifier" to recognize how to use the variable. For example, remember the glass of refreshment? You could tell what is in the glass by the color of the drink. The color is the identifier. As you might guess, if we need both the data and an identifier this can best be represented with a structure.

Before trying to demonstrate this technique, I should point out one more thing about unions — the members of a union do not have to require the same amount of space. In our first example it was important to ensure that the members used the same amount of space, but it is not mandatory. The space that the compiler sets aside for a union will be equal to the largest member of the union.

As I mentioned at the beginning, it is difficult to come up with an example that everyone can understand, so keep this in mind while you look at the example. Thus program lists a few

things that might be found in your house and provides some type of description for each item. To make things simple we will limit our items to furniture and electronics. After an item has been defined, the variable is passed to a function that will print the variable to the screen. Although this is a contrived example, I hope it demonstrates to you how the union can be used in your program.

```
/* Program INVENTORY.C */  
/* This program will demonstrate the use of the union */  
/* Several household items will be placed into the union */  
/* and then printed to the screen */  
  
#include <stdio.h>  
#define FURNITURE 1  
#define ELECTRONIC 2  
  
struct FURN {  
    char type; /* furniture electronic */  
    char name[20]; /* type indicator */  
    char color[10]; /* name of item */  
    float value; /* color of item */  
}; /* value of item */  
  
struct ELEC {  
    char type; /* electronic definition */  
    char name[20]; /* type indicator */  
    long serial; /* name of item */  
    float value; /* serial number */  
}; /* value of item */  
  
union ITEM {  
    char type; /* Household Item union */  
    struct FURN furn; /* type indicator */  
    struct ELEC elec; /* furniture structure */  
}; /* electronic structure */  
  
main()  
{  
    short count; /* start of program */  
    union ITEM item; /* count of item printed */  
    /* ITEM variable */  
  
    printf("Printing household items \n");  
  
    setmem(&item, sizeof(item), 0); /* Clear structure */  
    item.type = FURNITURE; /* Furniture */  
    strcpy(item.furn.name, "Sofa");  
    strcpy(item.furn.color, "Brown");  
    item.furn.value = 600.00;  
    list_item(&item); /* call the printing function */  
  
    setmem(&item, sizeof(item), 0); /* Clear structure */  
    item.type = ELECTRONIC; /* Electronics */  
    strcpy(item.elec.name, "TV");  
    item.elec.serial = 1234567;  
    item.elec.value = 350.00;  
    list_item(&item); /* call the printing function */  
  
    setmem(&item, sizeof(item), 0); /* Clear structure */  
    item.type = FURNITURE; /* Furniture */  
    strcpy(item.furn.name, "Table");  
    strcpy(item.furn.color, "Glass");  
    item.furn.value = 1200.00;  
    list_item(&item); /* call the printing function */  
  
    setmem(&item, sizeof(item), 0); /* Clear structure */  
    item.type = ELECTRONIC; /* Electronics */  
    strcpy(item.elec.name, "Stereo");  
    item.elec.serial = 222333;  
    item.elec.value = 450.00;  
    list_item(&item); /* call the printing function */  
  
    setmem(&item, sizeof(item), 0); /* Clear structure */  
    item.type = FURNITURE; /* Furniture */  
    strcpy(item.furn.name, "Desk");  
    strcpy(item.furn.color, "White");  
    item.furn.value = 175.00;  
    list_item(&item); /* call the printing function */  
  
    setmem(&item, sizeof(item), 0); /* Clear structure */  
    item.type = FURNITURE; /* Furniture */  
    strcpy(item.furn.name, "Book Shelf");  
    strcpy(item.furn.color, "Black");  
    item.furn.value = 226.00;  
    list_item(&item); /* call the printing function */  
  
    setmem(&item, sizeof(item), 0); /* Clear structure */  
    item.type = ELECTRONIC; /* Electronics */  
    strcpy(item.elec.name, "Amiga");  
    item.elec.serial = 777123;
```

(continued on page 104)



# A Common User Interface for the Amiga

by James Bayless

President—New Horizons Software, Inc.

A lot of discussion has taken place in the Amiga community about adopting a common user interface for Amiga programs. Often this debate has been quite heated, but sadly it has produced no concrete plans for adopting a common interface. On the one hand are those who argue that a common user interface would greatly benefit users of Amiga software, while others say that such an interface would stifle "creativity" in software development.

Furthermore, even developers who feel there should be a common user interface are divided into many different camps, unable to agree even on basic principles. Many even confuse the concept of a "common user interface" with things like file requesters or color palette requesters, things which are properly classified as only specific applications of a common user interface. Complicating this problem is the fact that no one at Commodore has either the qualifications or interest in researching and developing a properly designed user interface. The management at Commodore is apparently only interested in selling as many machines as they can in as short a time as possible, with little regard to the long term advantages of implementing a common user interface.

In this article I hope to both convey the very real benefits of adopting a common user interface, as well as to propose some preliminary guidelines for the interface. I will show that a properly designed user interface will benefit users without stifling creativity, and therefore there is no reasonable justification to not adopt one. While the problems of getting other companies, and Commodore itself, to adopt guidelines such as these is not directly addressed, such a change can only occur if there is enough user demand. This, ultimately, is the real reason for this article.

## **The Pros and Cons of a Common User Interface**

When writing computer programs, as when writing books or articles, the single most important tenet to adhere to is to always consider the intended audience. In the Amiga market, this audience has so far been primarily limited to computer hackers and the adventurous. Both of these classes of users understand the potential of the Amiga hardware and system software, and are willing to put up with difficult to use and poorly designed software to tap this potential. Business users, however, have for the most part avoided the Amiga. By "business users" I mean not just people in large corporations, but those in small and home-operated business as well—a substantial market segment. While some Amiga developers may be happy with limiting themselves to non-business users, many (myself included) feel that for the machine to succeed like we think it can, it must become a viable business computer. A well thought out, common user interface between programs can help substantially to penetrate this market. There are several very good reasons for adopting a common user interface:

### **1. A common interface makes it easy for a person to start using a new program**

Engineers and hackers like new and completely different things (otherwise why would they be involved in a fast moving field like computers?). Users, those who only want to use computers to accomplish a given task, and especially business users, want ease-of-use and ease-of-learning rather than feature-upon-feature. If every Amiga productivity program had a common user interface, the user would not face a totally different way of doing things every time they purchased a new program.

As an example of what a common user interface can do, consider the automobile. If every auto manufacturer insisted upon doing things differently than their competition—one makes the driver sit in the back, while another makes you steer with your feet and brake with your hands—then automobiles would not be as nearly as commonplace (or as useful) as they are today. Once a person learns to drive one car, they can drive just about any other car on the market. The performance may not be the same, but the operation is.

### **2. A common interface gives consistency from one application to another.**

Consistency from one program to another increases the perceived ease-of-use of all programs—once a user learns the ins and outs of the user interface say, on a word processor, they can jump right in to the meat of other programs, such as spreadsheets or database systems. And they will think that these other applications are easy to use even though they may be very complicated programs—simply because they already learned part of the new program's operation by using other programs that had a similar user interface.

### **3. A common interface gives a polished appearance to programs without each programmer doing the work.**

Appearance is very important to many users. Would you buy an ugly car instead of a sexy one simply because the ugly one had a few more horsepower? Most people wouldn't. The Amiga sorely lacks a polished appearance in its application software. In fact, the one interface on the Amiga that nearly all users have to interact with daily, the Workbench, is one of the worst abusers

(continued)



**NAG<sup>®</sup>  
PLUS 3.0**

## Schedule Assistant

Software for the  
AMIGA™ computer

### Voice and Sound Reminders -

Create your own or use NAG PLUS library.

### Perpetual Calendar -Enter 99 events per day.

### Auto Dialer -Connects YOU, or your modem.

### AREXX Port -Commands any timed event or action.

### Notepad -Click on any word to open text editor.

### Print -One click prints any file or appointment list.

Suggested Retail \$79.95. Ask your dealer or contact:



17730 15th Avenue N.E.  
Suite 223  
Seattle, Washington 98155  
(206) 363-6417

in this regard. The Amiga needs to get rid of the "hacker" image, which is caused primarily by having the engineers design the user interface. While home users don't mind the "gadgety" interface that much—some of them even like it since it gives them a sense of accomplishment when they master a strange new program—business users will generally go for appearance when given a choice. By having a common user interface, every program on the Amiga could benefit from a well thought out and polished appearance. However, the interface must not just be nice looking, it must be well thought out. Subtle details are important, as they give the program the right "feel."

*There are, of course, many reasons against a common user interface:*

### 1. *The Amiga is an enthusiast's machine, and shouldn't be limited by a common user interface.*

This is true, the Amiga is currently a machine that appeals primarily to enthusiasts. However, by adopting a common user interface we can expand the market. Of course, if the developer feels that the common user interface is not appropriate for their intended audience (music software comes to mind), he or she would of course be free not to use it. Therefore, by adopting a common user interface no market segment will be left behind, but new ones could be added.

### 2. *Computers with a standard user interface are boring.*

Again, let's look at the example of automobiles. While a VW Bug may be boring, a Ferrari certainly isn't. Yet they both have the same "user interface." A person who learned to drive in a VW Bug could easily climb into a Ferrari and drive away. Obviously then, a common user interface does not make a program—or a machine—boring. It is the program's creativity (or lack thereof) that determines how exciting it is. A common user interface simply makes it easier for users to learn and use these creative and powerful new products.

So a common user interface would neither hinder the Amiga's development nor eliminate an existing market

may be something completely different. But they are currently the expected way of doing things—just as the expected way of operating an automobile is by steering with your hands on a large wheel, having the clutch to the left, brake in the middle, and throttle on the right, and so on.

Don't be afraid to build on what has already been done! Too many people are defensive about the Amiga when it is compared to other machines. Many have the attitude of "If I had wanted a Mac (interface) I would have bought a Mac," or "The Amiga is better than the Mac, so it has to be different." Again, this is a lot like Ferrari saying "Ferraris are better than VWs, so they must be completely different." Such an attitude would be just as suicidal for Ferrari as it has been for the Amiga.

This does not mean that you can't improve on other graphically based interfaces. However it does mean that you should look hard at what has already been developed, what works and what doesn't, and what people expect an interface to look like. Then build from there. Don't re-invent the wheel!

With this in mind, the ideas presented in this article use as their starting point the methods used in the Macintosh computer. The choice of the Macintosh style is made for several reasons: 1) The Macintosh user interface is very well thought out, and is becoming a kind of standard against which others are judged—companies like Microsoft and IBM also use the Macintosh style as a baseline. 2) To

## New Store!

## Fisher's Computers & Software

Authorized Amiga Dealer

8005 Archibald  
Rancho Cucamonga, CA 91730  
(714) 987-1662

*Serving  
The Inland Empire*

segment. However, it can improve its image (and usefulness) in the business market (both small and large) and therefore bring knowledge, appreciation, and sales of this machine to this large and fast growing market.

For a small computer to be successful, the software's ultimate target should be to be people who want to accomplish real work, and are looking for something that will make learning and using their computer easier. These people are not interested in the latest technology unless it decreases the amount of work they do, or reduces the time they currently spend working. Pull down menus, windows, and so on may not be the best way to do things—the best way

## 1988 TAXES

Calculate your 1988 Taxes with  
Quality Business Systems' **TAX PLAN**  
**FAST, EASY, ACCURATE**

- Prints Forms
- Audits Taxes
- Multiple Returns
- 20 Forms and Schedules
- Saves to Disk
- Forecast 1989 Taxes

Commodore-Amiga  
ONLY \$34.95

Add \$4 S/H, MA add 5% Sales Tax. Check, M.O.,  
Credit Cards

**Quality Business Systems**  
P.O. Box 805, Hudson, MA 01749  
1-800-225-5800



many non-technical computer users, the name Macintosh has become synonymous with ease-of-use. 3) Most importantly, the Macintosh way is currently (rightly or wrongly) perceived to be the correct way of implementing a user interface.

The intention, however, is not to duplicate the Macintosh, but rather to build upon the ideas presented there—to extend them and improve upon them.

### ***Suggested Elements of a Common User Interface***

When deciding upon the elements of a common user interface, thought should go into not only the look, but the function—what each element does. Any properly designed user interface must have a reason for everything it has or does. The following sections offer some suggestions for the elements to be included in a common Amiga user interface. Many are simply style recommendations, while others are specific recommendations for routines to be part of a sharable software library.

These recommendations are aimed at application programs and utilities. Programs like games shouldn't be concerned with these items, since a person engrossed in a game is not thinking about using a computer—they want to be totally immersed in the world of the game.

### ***Text Editing***

While the block cursor may be appropriate for text editing when all characters are the same width (i.e. when using mono-spaced fonts only), when dealing with proportionally spaced fonts a block cursor is virtually unusable. A better approach is an insertion point—a vertical bar one or two pixels wide, possibly blinking to help the user find it, that goes between characters and indicates where the next typed character will be inserted. While the use of an insertion point will make all text entry “insert mode” only, this is not a problem, since the concept of “overwrite mode” is not usable with proportionally spaced text any more than a block cursor is.

Most operations on a computer are best carried out in an “object-action” approach; the first user selects the object, then chooses the action to apply to that object. Text editing is no different. The objects in this case would be pieces of selected text. For example, to change

text to boldface the user would first select the text, then choose “bold” from a menu. There should be several methods available for moving the insertion point and selecting text, using either the mouse or the keyboard:

#### **Single click**

Place insertion point at location clicked. No text is selected (any previous selection is de-selected).

#### **Double click**

Select the word that is under the mouse pointer.

#### **Triple click**

Select the complete line that is under the mouse pointer.

#### **Click and Drag**

Continuously select text between the original click and the current mouse position. Selection is completed by releasing the mouse button. If the user double clicks the mouse before starting the drag, words will be selected; if the user triple clicks, lines will be selected.

#### **Keyboard**

If a cursor key is pressed move the insertion point in the direction indicated. If no modifier keys are held down, such as the shift, alt, or control keys, then move left or right by one character, up or down by one line. If the shift key is held down, move by the next logically larger unit: by word left or right and by screen-full up or down. If the alt key is held down, move by the largest logical unit: to the beginning or end of the current line, or to the beginning or end of the document.

To select text when using the keyboard, the user should simply hold down a modifier key to select text as if the mouse were being dragged across the text. The most logical modifier key to use would be the left and/or right Amiga keys. Unfortunately the wizards at Amiga have designed the system to use Amiga-Cursor key combinations to move the mouse pointer (How many people actually use this?), there is no simple way for a program to override this.

The only remaining modifier key available is the control key. Therefore, if the user presses the control key while pressing a cursor key (with or without also pressing the shift or alt keys) the text should be selected as the insertion point moves over it.

(continued)

*Willkommen  
Bienvenue  
Welcome*

**Alle ausländische  
Computergeschäfte /  
Zeitschriftenhändler**

Willkommen in der erstaunlichen Welt von Amazing Computing™ - eine von Amerikas führenden Zeitschriften für Commodore Amiga Benutzer. Amazing Computing™ war die erste Zeitschrift, die über das CLI berichtete, die erste Zeitschrift mit 1 Meg Amiga Hardware Projekt Verbesserung und die erste Zeitschrift, die seriöse Programmhilfsassistenten anbot.

Wenn Sie mehr Information über die Stelle des Amazing Verkäufers oder Amazing Werbefachmanns haben möchten, kontaktieren Sie bitte:

Marie A. Raymond  
International Coordinator  
PIM Publications  
P.O. Box 869  
Fall River, MA 02722  
(508) 678-4200  
(800) 345-3360  
FAX: (508) 675-6002

**Amazing**  
COMPUTING™  
Your Original AMIGA  
Monthly Resource



If the user types something when text is selected, the selected text should be replaced by the newly typed text. This gives a pseudo-overwrite mode. Ideally there should be an "undo" command available to undo the operation if it was done accidentally. Other operations that could be applied to selected text are Cut, Copy, Paste, and Erase operations.

### Screens

Custom screens should always have a drag bar and front/back gadgets, since on a multitasking machine like the Amiga the user should always be able to return to the Workbench screen and access other programs or files.

For a similar reason, a program should never remove the Workbench screen, except for a short time and only to complete a user-requested command—more memory is needed for creating and printing a large bit-map. If a program does remove the Workbench screen temporarily, it should replace it at the earliest convenience.

Another option, though less automatic, is to let the user choose whether or not to have the Workbench screen available. This has the advantage of being easier to program, but has the disadvantage of not letting the program use the extra memory of the Workbench screen when it needs it; it must specifically request the user to remove the Workbench screen and try the command again. However, if a program does take this route, then the default condition should be to have the Workbench screen visible.

When a program opens a high resolution (interlaced) screen, using the standard Topaz-8 font results in tiny, hard to read characters. Therefore, the program should also use a larger font for menu and requester displays. Since 99% of what people read is proportionally spaced, using a proportionally-spaced font would be preferable; but using a proportionally spaced font in menus and requesters is difficult with the current design of Intuition. (As far as I know, ProWrite is the only commercial program to use a proportionally spaced font for menus and requesters). The font Topaz-11, while not proportionally spaced, is tall enough to be usable in these high resolution displays.

### Windows

Windows are the basic area for user interaction with a program. However, as simple as they may seem, several factors must be considered when designing windows. Windows should open with the title bar below the screen's title bar to give the user access to the screen's drag bar and its front and back gadgets.

A window should have a re-sizing gadget only if it makes sense to be able to re-size the window. A utility program that has objects and gadgets in its window at fixed position should therefore not have a re-sizing gadget. On the other hand, application programs that can display variable amounts of information in a window (such as a word processor or spreadsheet) should have re-sizing gadgets. In simpler terms, a window shouldn't have a re-sizing gadget unless there is a real reason to have one.

Windows that don't have re-sizing gadgets should initially appear centered in the screen. Windows with re-sizing gadgets should (unless there are compelling reasons to do otherwise) open the full width and height of the screen, minus the size of the screen's drag bar.

In general, utility programs or programs that only open one window should appear in the Workbench screen. Programs that open multiple windows should open their own screens. This is to avoid cluttering up the Workbench screen with too many windows. A program that opens up its own screen should also open up a background window in that screen and attach at least a minimal menu strip to that window so the user is never confused as to why he can't see any menus when trying to pull them down. A program that does open a background window should fill the window with a color or pattern to indicate the existence of the background window and its boundaries.

### Menus

There should be a standard appearance for all normal (text and simple image) menus. Since most people read left-to-right, all menu items should be aligned on the same left margin location in the menu. Additionally, since some menu items will have check marks to their left, all menu items should have space for a check mark to their left to ensure consistent left margin locations.

Common operations should be grouped together in a menu, separated from other menu commands by a dotted line. This will help users to find the menu items they want without reading the entire list of menu commands.

Unless there are compelling reasons to do otherwise, all applications should have a minimum standard set of menus for loading, saving, printing and basic editing operations. This will greatly increase the user's ability to switch from one program to another, as it allows them to accomplish these basic operations without having to remember different procedures for each program. A suggested set of minimum menu items is:

#### Project menu:

```
New
Open
Close
-----
Save
Save As
Revert
-----
Page Setup (or Print As)
Print
-----
Quit
```

#### Edit menu:

```
Undo (optional)
-----
Cut
Copy
Paste
Erase
-----
Select All
```

There should be a standard location for sub-menu. Ideally, since people read left-to-right, it should appear at the far right of the menu item it is attached to. Sub-menus should not obscure any portion of the text of the menu items underneath it.

Unfortunately, having a sub-menu located to the far right of the menu makes it difficult to select items on the sub-menus. The user has to click on the menu, drag vertically down to the item that has a sub-menu, drag horizontally to the sub-menu, then vertically again to the desired sub-menu item. A better way would be to let the user drag diagonally from the menu item to the sub-menu item rather than having them drag horizontally then vertically. Unfortunately, diagonal dragging to sub-menu items is something that would have to be implemented in Intuition itself.



Since the Amiga doesn't let you drag diagonally to a sub-menu item, it is desirable to have the sub-menu as close to the menu item as possible. Since every menu item should have space to its left for a check mark, and since a menu item that has a sub-menu should never need to be checked, the logical place for a sub-menu is to the left of the menu item, overlapping in the space reserved for the check mark. This will allow quick access to the sub-menu without obscuring any portion of the text of the menu items alongside or below the sub-menu. The drawback, of course, is that now the sub-menu will occur to the left of the menu item it is attached to, an arrangement that is not natural for left-to-right reading languages.

### Gadgets

Gadgets should have meaningful names for the specific operations they perform. Don't call them "this gadget" and "that gadget." If the gadget looks like an arrow and you click on it to scroll something, call it a "scroll arrow," not a "scroll gadget." If it is a long horizontal bar and you click and drag on it to drag something (like a window), call it a "drag bar," not a "drag gadget." This simple naming convention makes it more clear to the user that the different objects perform completely different and unrelated operations.

The current convention of calling them all "gadget" came about because the programmer accesses them in the same manner. To the user, though, they are not similar in either appearance or operation. Since the goal of any well-designed user interface is to make things clear from a user's point of view, the gadget names should reflect the way the user sees them.

Different classes of gadgets that perform different types of operations should have distinctive appearances so the user can quickly tell what type of operation will be performed by clicking on a particular gadget. Among the classes of gadgets there are:

#### "Perform this action" gadget.

Should be called a "Push Button" or simply a "Button." They should be used only to initiate an action (such as proceed or cancel a requester's operation), and should never be used to simply turn on or off an option. They will most commonly occur in requesters. (OK and Cancel buttons for example).

#### "Select this option" gadget.

There can be several types, depending upon number and type of options to be chosen from:

##### "Check Box"

Used to turn on or off a binary option.

##### "Option Button" or "Radio Button"

Used to select one from a small list of mutually exclusive options, they should be grouped to make obvious what constitutes the group of options. If only an on/off selection is needed, the check box should be used.

##### "Pop Up Menu"

Used to select one of a variable number of options. Unfortunately this type of gadget cannot be easily implemented under the current design of Intuition.

##### "List Box"

Used to select one of a possibly large and variable number of options, the List Box consists of an area for displaying a fixed number of options, and a scroll bar (consisting of both a scroll box and scroll arrows) to scroll among the entire list. Can be implemented as either a mutually-exclusive selection—selecting one option de-selects the previous selection, as a cumulative selection, or both. When implemented as both, a simple click should act as a mutually-exclusive selection, a shift-click should act like a cumulative selection by adding this new selection to previously selected items and a shift-click should remove the item from the list of selections.

#### "Enter text here" gadget

Should be called a "Text Box." Used to enter textual information. Should not be called "String Gadget", since to the casual user the term "string" has absolutely no connection to the process of typing text.

#### Others

Several other types of specialized gadgets may be useful in application programs. The most common are scroll bars and active images (i.e. pictures or icons that perform some action when clicked on).

#### Scroll Bars

Consists of scroll box (proportional gadget) and scroll arrows. Scroll arrows should scroll continuously when mouse

(continued)

*Willkommen*

*Bienvenue*

*Welcome*

### Magasins d'ordinateurs à l'Etranger et Marchands de Magazines

Bienvenue au monde stupéfiant de Amazing Computing™ - un des magazines les plus importants pour les utilisateurs du Commodore Amiga aux Etats-Unis. Amazing Computing™ fut le premier magazine à documenter CLI, le premier magazine ayant développé le projet d'extension mémoire 1 Mega pour Amiga, et le premier magazine à offrir une assistance sérieuse de programmation.

Ne refusez pas à cette large partie de votre clientèle anglophone, ce magazine qui leur fournira la plus complète information sur l'Amiga. Venez et devenez membre de notre monde stupéfiant comme marchand ou annonceur.

Pour devenir un Amazing Dealer ou un Amazing Advertiser, nous vous prions de vous mettre en contact avec:

Marie A. Raymond  
Coordinatrice  
Internationale  
PiM Publications  
P.O. Box 869  
Fall River, MA 02722  
(508) 678-4200  
(800) 345-3360  
FAX: (508) 675-6002

**Amazing**  
COMPUTING  
Your Original AMIGA  
Monthly Resource



button is held down over the gadget (i.e. only while the scroll arrow is selected). Scroll boxes should not continuously scroll when dragged. This looks cute when scrolling a small list of items, it becomes distracting when scrolling a large list, and becomes unusable when scrolling through a large multi-page document. Therefore since it cannot be used for large documents, and shouldn't be used for long lists of items, then for consistency it shouldn't be used even for small lists. If the user wants to see the list of items scroll by continuously they should click and hold down on the appropriate scroll arrow.

### **Application-defined gadgets**

There should be the capability to include either simple non-standard gadget definitions, e.g. "use this image for a on/off gadget," or to include complete gadget definitions, e.g. "use this full gadget structure." It should be possible to mix these custom defined gadgets with standard ones and have everything handled properly.

### **Requesters**

Requesters should be used only for "modal" states—when the user is in a mode where the program needs information from the user before proceeding with a requested operation. A requester should not be opened unless it is absolutely necessary (normal user interaction should be handled with a window).

A requester should be centered in the window it appears in (but should always be below the window's title bar, if any). If a requester opens its own window, the window should be centered on the screen and it should not have a re-sizing gadget in it. Since requesters represent a "modal" state, they should not be able to be reduced in size and ignored. (Note that this means "System Requests" shouldn't have re-sizing gadgets in their windows).

There should be standard style for requester borders. This will tell users they are in a "modal" state and that they must satisfy the requester before they can proceed.

There should be a simple way of including in the requester not just gadgets, but static images and text for labeling and identification. A static image or text item produces no response or action when the user clicks on it.

Color should be used sparingly—in pictures and possibly for emphasis of particular items. Too much color is confusing and therefore counterproductive.

Nearly all requesters should include an "OK" or "Proceed" button and a "Cancel" button. The "OK" button should have some special highlighting or emphasis, and pressing the RETURN or ENTER key should be equivalent to pressing the "OK" button (and ideally there should be some visual feedback when the key is pressed). Similarly, pressing the ESC key should be equivalent to pressing the "Cancel" button.

### **Miscellaneous**

There should be standard, sharable images for requester pictures (icons), scroll arrows, and mouse pointer images. This will maintain consistent appearance of familiar items from one program to another. Typical mouse pointer images would be an arrow pointing to the upper left, a "busy" pointer, and an "I Beam" pointer used when placing an insertion point in text.

There should be a convention for assigning keyboard equivalents for common operations (such as Cut, Copy, and Paste). Other keyboard equivalents should be logically assigned (For example, if F7 means "Exit" then shift-F7 should not mean "Print"). Again, this will allow users to accomplish the same basic tasks in an identical manner in different programs.

### **A Standard File Requester**

Contrary to what many users, and sadly, many developers think, file requesters alone do not comprise user interfaces. A standard file requester does not in and of itself constitute a common user interface, it merely a small part of the overall picture. Far too much argument and debate have gone into discussing the merits and drawbacks of one file requester over another, while the real problem of a consistent and easy-to-use interface for all other aspects of the Amiga's operation—screens, windows, menus, gadgets, and "non-file" requesters—have been completely overlooked. Unfortunately, these other elements are what a properly designed file requester is built from. Without the proper foundation of a well thought out window, gadget, and requester design, it

is difficult to design a proper file requester. Even accomplishing it without this foundation, will leave you with the file requester and no foundation on which to build other items.

### **There are several requirements for a basic file requester:**

1. It should use the elements of a generalized gadget and requester design to maintain consistency with other gadgets and requester appearances in the program.
2. It should be callable from the program by a single function call with a minimum of parameters. This eliminates the class of "intelligent" file requesters which attempt to work around the slowness of AmigaDOS by cacheing the directory contents, since this would at the minimum require an additional function call at program termination to free memory allocated for the directory cache. Furthermore, such a directory caching and other DOS speedups are more appropriately handled by improvements to DOS itself, not by having every program attempt to make their own work-arounds.
3. It should be customizable to varying degrees. Three levels of customization and call-back would give just about any amount of control a calling program could desire:
  - a. Optional call-back on each file to have the calling program determine if the file should be listed.
  - b. Optional call-back on each gadget down/up operation to allow calling program to handle requester gadgets in its own manner, or to handle gadgets that it has added to the requester display itself.
  - c. The ability to substitute a completely different list of gadgets in the requester. If the gadgets are the same type as the requester's expected gadgets, then this could serve to simply change the imagery of the requester without changing the overall operation. Additionally, if the calling program requests call-back on each gadget operation, then it could conceivably substitute any gadget list and have complete control over both the requester's appearance and operation.



4. There should be different routines and displays for "Open" file and "Save" file operations. These two operations are fundamentally different to the user, one has users selecting files to load from pre-existing files. The other operation has users provide a directory path and name for a file to create or overwrite, and so they should have different requester presentations, to allow users to know at a glance if they are trying to load a file or save one.

5. A file requester should under no circumstances display ".info" files in its file list. Furthermore, since many utilities create files that begin with a period, it is desirable to avoid displaying files whose names begin with a period.

### **Roadblocks to User Interface Improvements on the Amiga**

Several areas of the Amiga system design inhibit development new interface ideas on the Amiga. In just about all cases the inhibiting factor is Intuition. Hopefully, these difficulties will be addressed in future improvements to Intuition.

First of all, Intuition makes it difficult to create the elements of a user interface. There is no "high level" support. To create a gadget, you must define the actual structure that contains every detail about the gadget's appearance, location, highlighting, and system handling. There is no way to say, for example, "give me a gadget with a push button image and operation with this title and at this location." Instead, you must create the actual gadget structure, specify explicitly the highlighting mode, create the border or image associated with the gadget's appearance, and link it to the proper requester or window.

The same is true for creating menus and requesters. True, many people either have created tools themselves or use tools others have created that will handle this to one degree or another. However, the fact that so many people have had to create these tools indicates that something is missing in the Amiga itself. If these routines were collected into a common library, it would not only save future developers from repeating other developer's work, but it would give the

programs using it the consistent appearance and operation that is the hallmark of an easy to use system.

The other major obstacle presented by Intuition is more subtle, and unfortunately cannot be currently circumvented within the confines of Intuition. That problem is the lack of "low level" support in Intuition. There is no way to "get under the hood" so to speak, you cannot create new user interface elements that are not part of the limited set already provided in Intuition itself. What is needed is a way to extend Intuition in a system-supported manner, a way to create new types of gadgets, or menus, or windows directly. This could be implemented if Intuition were designed with a "call-back" type of interface to the program—it provides standard "drivers" for the display and handling of standard gadgets, requesters, windows, menus, and so on, but would also give you the capability to define your own "drivers" and hook them into Intuition. This area needs to be addressed in any future versions of Intuition.

### **Proposal: A "toolbox.library" Extension to Intuition**

The first step in creating a common user interface for the Amiga is a "toolbox.library" that contains routines that would automatically create (and dispose of) standard appearing requesters, menu strips, gadget lists, and so on, given minimal information about what these structures and lists should contain. This would reduce code complexity by eliminating the need to create and compile the full Intuition structures into the program. And it would reduce program size, since only the minimal list descriptions would be part of the load module, the actual structures would be created and freed dynamically, and give a consistent appearance and operation to all programs.

Such a library should include routines for handling common gadget, requester, and menu operations, such as modifying a menu's contents after it has been created, tracking when the mouse is over a selected gadget (useful for scroll arrows), handling basic requester operations, and so on. It should also include shared images for mouse pointers of various types, scroll arrows, and common requester icons.

(continued)

*Willkommen*

*Bienvenue*

*Welcome*

### **All Foreign Computer Stores / Magazine Dealers**

Welcome to the amazing world of Amazing Computing™ - one of America's leading magazines for Commodore Amiga users. Amazing Computing™ was the first magazine to document CLI, the first magazine with a 1 Meg Amiga upgrade hardware project, and the first magazine to offer serious programming assistance.

Don't deny your large English-speaking audience the kind of magazine that will provide them with complete information on the Amiga. Come and join our Amazing world as either a dealer or an advertiser.

For details on becoming an Amazing Dealer or an Amazing Advertiser, please contact:

Marie A. Raymond  
International Coordinator  
PIM Publications  
P.O. Box 869  
Fall River, MA 02722  
(508) 678-4200  
(800) 345-3360  
FAX: (508) 675-6002

**Amazing**  
COMPUTING  
Your Original AMIGA  
Monthly Resource



## 68000 DISASSEMBLY



### ANNOUNCING... **DSM** VERSION 1.0d

DSM is a full-featured disassembler for the Amiga. Check out these features and you'll see why programmers agree, "DSM is the best disassembler currently available for the Amiga, bar none."

- DSM disassembles any Amiga program that does not use overlays.
- DSM produces assembler-ready output that is 100% compatible with the Amiga assembler, *assem*. Excellent results can also be obtained using the C.A.P.E. assembler from Inovatronics.
- DSM features true text detection.
- Improved disassembly is supported using the built-in expert system mode.

If your local Amiga dealer doesn't carry DSM, send check or money order to:

OTG Software  
200 West 7th Street Suite 618  
Fort Worth, TX 76102

TX residents add 7.25% sales tax. Price \$67.50

This library could include routines for the "Load" and "Save" file requesters as detailed previously. However, since a standard file requester is not part of a "toolbox" used to build gadgets, menus, and requesters, but simply uses these routines, these file requester routines would be more appropriately placed in a library of their own. This "stdfile.library" would of course use the routines in toolbox.library to create and manage its gadgets and requesters, but as a separate module developers would be free to use the toolbox.library without also loading these file requester routines. Similarly, as other commonly needed requester routines were created, they would be put in a library of their own (such as a "palette.library" to handle a common color palette requester, and so on).

What should not be in the toolbox.library is anything not directly related to the "higher level" above Intuition. Anything that does not directly relate to user interface and Intuition enhancements should be kept out of the library; the point of a system library is to collect common routines with a well focused purpose that application

programs can choose to take advantage of. If there are too many "extra baggage" routines then many developers will choose not to use it because of RAM considerations. Therefore, a system library should not be a conglomeration of unrelated routine, like DOS enhancements, file requesters, memory management, and program loading and execution. Such an approach is poorly thought out and poorly designed. DOS enhancements should be in DOS, memory management in Exec, and so on.

At New Horizons Software we have developed a linkable-library that includes all of the features discussed in this article. It forms the core of the user interface presented in all of our Amiga products. This library could conceivably be developed into a sharable library, and therefore could serve as a starting point for a discussion of routines that should be part of such a toolbox.

#### **Final Suggestions**

Even lacking a "toolbox.library" to support a common user interface, several things can be done to improve the usability of application programs and image of the Amiga computer in general, at least among the non-computer literate crowd:

#### **Speak to the average person.**

Don't use buzz words when a more conventional word or words will work. Some examples: Use "Memory" instead of "RAM," use "Operating System" or "Kickstart" instead of "ROM," use "Graphics Memory" instead of "CHIP RAM," and "Expansion Memory" instead of "FAST RAM." The latter terms are fine for developers and hackers, but the average person who only wants to use their computer shouldn't have to decipher "FAST RAM" (and especially not "SLOW FAST RAM"). Nobody needs to know about cam shafts and pistons to operate an auto, and computers should be no different.

Don't use "techie" talk. If you can't explain it in everyday terms, you are doing something wrong—it is either too complicated (and should be broken up into simpler steps) or unnecessary. Of course if you are designing your program for "enthusiasts" only then feel free to use all the techie terms you want.

#### **Don't do something because it is "cute."**

Everything in the program should be well thought out; features (or gimmicks) should not be added unless there is a clear reason for them. And if you do add more features, try to add them so that the program feels logically consistent. A feature should not have an obviously "tacked-on" appearance.

#### **Write serious looking programs.**

"Serious" doesn't mean it can't be attractive. But it should not be "cute." Don't try to impress people (or yourself) with your programming wizardry by making things too fancy or adding too many functions. Users (assuming that they have a basic knowledge of the program's function) should be able to figure out at least 70% of what they need to do without having to look at the manual.

Use more serious sounding names for the user interface items. The Amiga already has a bad enough image program without the stigma of "gadgets." If an auto manufacturer called the throttle in their auto a "go gadget," the brake the "stop gadget," the steering wheel the "turn gadget," and so on, then they would have an image problem too!

Amiga owners know what their machine is capable of. The software at the heart of the Amiga—the multitasking Exec, hardware, device drivers, and graphics system—are well designed and solid. But the aspects of the Amiga that people get their first impressions from—AmigaDOS, Intuition (and the "gadgety" user interfaces it inspires), and Workbench—are either not well thought out or just plain bad. And first impressions are hard to overcome.

If the Amiga is ever going to be accepted by the business user, it does not need another demo, it needs more real application programs—programs that are both powerful and easy to use. But most of all, it needs programs with a consistent user interface. The Macintosh has proven that a consistent user interface is a powerful asset to any computer, and even stodgy old IBM has seen the light. And while such a common user interface will not guarantee the Amiga's success, it will certainly help its chances.

•AC•





1&gt;

# The Command Line

*Become an ED Power User*

*by Rich Falconburg*

Several issues ago I introduced you to ED, the screen editor delivered on your Workbench disk. We only scratched the surface of ED's capabilities and for good reason. Learning to use a text editor is much like getting use to a wordprocessing program. Some text editors provide much of the same power found in commercial quality WP programs. If you prefer to use your WP program, be my guest. As long as it is capable of writing a text ONLY file (and most are) then there is no reason why you shouldn't use it. Be aware that most full fledged WP programs require more memory than your average text editor, valid concern if you want to test your modified script without exiting the editor and you have limited memory resources. Personal preference generally is the deciding factor in choosing the editor that fits your needs.

So what is the big difference between text editors and Wordprocessing programs? To be honest, the latest entries to the market narrow the distinction considerably. Not all that long ago even the big systems were limited to simple line editors. A line editor uses a variety of command driven combinations for allowing alterations to a file. Although you can display the file, (similar to TYPEing it) most operations are line by line. A few commands allow global changes (the entire file) but even this may be limited. Your Workbench disk also contains this type of editor: EDIT. I won't be covering it because the only place EDIT is really useful is on a text only terminal connected through the Serial port via AUXCLI.

ED provides most of the same features as EDI, but as a full screen editor it allows you to use the cursor keys and various Control key combinations to move around and make changes on a full page rather than line by line. However, ED is similar to those old line editors in that it is "command driven" to perform more elaborate functions. This is OK but if I wanted to be stuck with keyboard-only editing, I'd use one of those other brands of computers. There is no Intuition Menu or Mouse support in ED. Instead, you press the ESCape key to enter the "extended" mode and enter a sequence of keystrokes to perform the power functions.

If you wish to use EDIT or ED extensively, I suggest purchasing the AmigaDOS Manual from Bantam Books (\$24.95). This book covers both in detail and provides a handy reference to AmigaDOS in general.

## Using ED

The following key sequences operate in the full screen mode. Control key combinations are indicated by a ^ and the letter of the key to be pressed while holding down the CTRL key.

KEY	OPERATION
BACKSPACE	Delete characters to the left
DEL	Delete characters to the right
ESC	Enter the extended command mode
RETURN	Start a new line at the cursor
TAB	Not a true TAB. Moves cursor to next TAB position
Cursor Keys	Move in the designated direction
^A	Insert a new line below the cursor
^B	Delete the line the cursor is on and close the gap from below.
^D	Scroll down
^U	Scroll up
^E	Move the cursor to top or bottom of the screen.
^J	Move the cursor to the start or end of the line.
^R	Move the cursor to the end of the previous word.
^T	Move the cursor to the beginning of the next word.
^O	Delete word or spaces to the right.
^Y	Delete from the cursor to the end of the line.
^F	Change case of character under the cursor. (Repeats if held down)
^V	Refresh the screen
^G	Repeat last entered "extended" command line

(continued)



**D - Five Associates**

19 Crosby Drive  
Bedford, MA  
01730-0523

(617) 275-8892

Tired of the high cost of computer repairs?

→ **FLAT** Labor charges

→ **FREE** Estimates


→ **Warranty** work

Also:


1764 to 512K: \$61<sup>95</sup>

128 64K vdc RAM: \$40<sup>00</sup>


NEW: C=1902 conversion to RGB-I: \$40<sup>00</sup>



Commodore PC-10




C-64/128 & peripherals



Amiga 1000

\* plus parts and sales tax

Authorized Commodore Service Center



### Extended ED Commands

Pressing the ESC key will cause ED to display an asterisk (\*) at the bottom left corner of the window. This is the "extended" mode command prompt. At this point you may use the following commands. The delimiter shown is the slash character (/) but this may be any special character that does not conflict with the string being entered and is not a space, semi-colon, or bracket.

## Cursor Movements

COMMAND	OPERATION
T	Move to the top of the file
B	Move to the bottom of the file
Mn	Move to line n.
N & P	Move to the beginning of the Next or Previous line.
CL & CR	Move the cursor left (CL) or right (CR) one character.
CS & CE	Move the cursor to the beginning (CS) or end (CE) of the line.

## Margins

COMMAND	OPERATION
SL & SR	Set the left or right margins
ST	Set the TAB distance
EX	Extend right margin (margin release one line only)

## Block Operations

COMMAND	OPERATION
BS & BE	Block Start and Block End. Place the cursor then enter either command.
DB	Delete Block (as defined with BS and BE)
IB	Insert Block - copies the block defined by BS and BE at the cursor.
SB	Show Block. Moves the cursor to the beginning of the defined block.
WB /file/	Write Block to file on disk

## Text Manipulation

COMMAND	OPERATION
A/string/	Insert string below current line
I/string/	Insert string as new line above cursor
IF/file/	Insert file below cursor
U	Undo changes just made on this line
LC & UC	Case sensitive (LC) or not (UC) on searches
F/string/	Find string in forward search
BF/string/	Find string in backward search
E/str1/str2/	Search for str1, replace with str2 (one occurrence only - see RP below)
EQ/str1/str2/	Search and replace with confirm
RP	Repeat command until an error occurs
D	Delete the current line.
S	Insert a carriage return
J	Join the current line with the next one

## File Operations

COMMAND	OPERATION
Q	Exit ED without saving text
X	Exit ED and save text to current filename (see the SH command below)
SA	Same as X without exiting ED
SA/file/	Save text as file (full path may be specified here and in WB above)

## Miscellaneous

COMMAND	OPERATION
SH	Show current file name, tab spacing, margins, block start and end, and buffer usage.



# Design digital logic hardware on the Amiga.

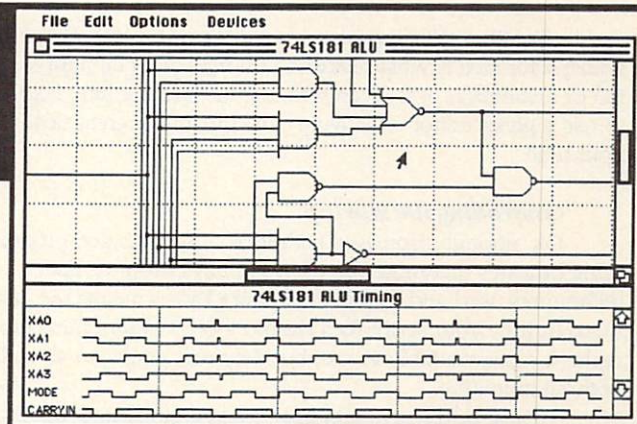
## LogicWorks

- simulates a "live" circuit on the screen
- instantly displays changes in inputs and device delays
- supports most common logic devices

**\$299.95**

Order Directly From Us or From Your Dealer

Trademarks: Amiga - Commodore Business Machines Inc. LogicWorks - Capilano Computing Systems Ltd.



**CAPILANO COMPUTING (604) 669-6343**

501-1168 Hamilton Street, Vancouver, B.C.  
Canada, V6B 2S2

Although ED is command driven, there is a lot of power from that command prompt. For example, any command may be repeated a given number of times by preceding the command with a number:

```
*3 IB
```

Or you could use the RP command to repeat the command until an error occurs:

```
*RP E/Tom/Bill/
```

replaces every occurrence of Tom with Bill and continues until the end of the file is reached.

Command chaining is where ED excels. To chain commands together, separate them with a semi-colon (;). Commands may be grouped together using parenthesis.

```
*T;BS;M8;BE;B;5 (B;IB)
```

This command string will locate the cursor at the top of the file, mark the beginning of a block, move the cursor to line 8, then mark the end of the block. The cursor will be positioned at the bottom of the file and the defined block will be added to the end of the file five times. As you can see, this provides some powerful capabilities that may not be found in other text editors. One problem with ED, is that you can't create a macro of this command string. Macros allow you to combine several commands into a single recallable command. This helps when you have repetitive steps to perform, especially since ED tends to be pretty slow.

To start ED you must supply the name of a file for ED to work with, otherwise it complains. You may also specify a larger buffer to work with if you are editing a large text file.

```
1> ED My_File SIZE 50000
```

This line will cause ED to read in My\_File if it exists and will allocate a buffer of 50,000 bytes for you to work with. If the file does not exist, a new one is created when you exit with X or save with SA.

What if you are terrible at remembering dumb little one or two letter commands and your convenient reference card is always hiding when you need it. Is there an easier way? Of course! After all, this is the Amiga!

On your Amiga Extras (1.2) disk in the Tools directory you will find a program titled MicroEMACS. It is a text editor based on a very popular version already running on a variety of other computers. This Amiga version supports standard pull down menus and the mouse. A documentation file is included with the program. I recommend that you print it out if you intend to use MicroEMACS extensively. MicroEMACS will perform most of the functions described for ED plus:

- Edit more than one file at a time
- Issue AmigaDOS commands or open a New CLI
- Insert special (non printing) characters
- Wrap words
- Special save features
- Macros and Key definitions
- Interlace option (for double the lines displayed)
- Cursor and Shift key combinations for quick moves
- Keyboard equivalents to menu options
- ...and more.

If you are a beginner at this text editing bit, I would suggest using MicroEMACS instead of ED. Purists will argue that other text editors available in the Public Domain may be even easier to use while providing as much and perhaps more power than MicroEMACS. These include DME by Matt Dillon (Fred Fish #134), UEdit by Rick Stiles (Fred Fish #121), and other versions of MicroEMACS.

(continued)



Try them all if you can. As I mentioned at the beginning, your preference is what matters. You may even be happier with ED or a variation of it. Even with menus and the like, learning to use a given editor effectively can consume a great deal of your time.

### Controlling the Startup

Ok, enough promises. Let's look at some batch programming quickies that might be useful to have around. The techniques used and format shown are by no means the last word in how your script files should look. I indent things to make it more readable to me, but they will work just as well without indenting.

If you don't have a battery backed clock and want to make sure that the system time gets set every time you boot, put this short script in the first part of your startup-sequence.

```
ECHO "Please enter the Date and Time (DD-MMM-YY HH:MM)"
DATE >NIL: ?
```

Pretty simple, huh? The question mark will cause DATE to print a usage prompt that we really don't care to see, so we send it to the NIL: device. Unfortunately, useful error-trapping with the DATE command is next to impossible.

Here's one that may be useful to someone. If you switch back and forth between a standard Workbench screen and an Interlace Workbench screen and use several script files to set up console windows, you may eventually run into trouble when creating a window. The size values can be a problem. This script and the one following it can help prevent problems by using two configuration files and providing a way to test for which one the system is using. To use it, you must copy your current system-configuration file to a new file. If you use the script shown "as-is", then the name of the new file should be "sysconf.lace" if you are currently using an interlace Workbench or "sysconf.norm" for a normal Workbench. After you copy the old file, run preferences and change the "Workbench Interlace" gadget to the opposite of what it is set to and save it. Now you can execute this script to change to the type of Workbench you wish to use and then reboot.

```
; Change configuration on BOOT disk
.key mode
IF <mode>use> EQ "use"
    ECHO "Usage: BOOTConf mode"
    ECHO " mode = NORM or LACE"
    QUIT
ENDIF
IF <mode> EQ "norm"
    IF EXISTS df0:devs/sysconf.norm
        RENAME df0:devs/system-configuration df0:devs/sysconf.lace
        RENAME df0:devs/sysconf.norm df0:devs/system-configuration
        ECHO "System reconfigured to NORMAL Workbench"
    ELSE
        ECHO "Unable to reconfigure system. The file needed"
        ECHO "for the requested mode is not available."
        LIST df0:devs/sys#?
    ENDIF
ENDIF
IF <mode> EQ "lace"
    IF EXISTS df0:devs/sysconf.lace
        RENAME df0:devs/system-configuration df0:devs/sysconf.norm
        RENAME df0:devs/sysconf.lace df0:devs/system-configuration
        ECHO "System reconfigured to INTERLACE Workbench"
    ELSE
        ECHO "Unable to reconfigure system. The file needed"
        ECHO "for the requested mode is not available."
        LIST df0:devs/sys#?
    ENDIF
ENDIF
```

```
ENDIF
ECHO "Reboot the Amiga to make the changes active."
    You can check for the type of Workbench in your startup-sequence
    and set up console windows using something like this:

IF EXISTS df0:devs/sysconf.norm
    ECHO "INTERLACE Workbench detected."
    SetFont topaz 11
ELSE
    ECHO "NORMAL Workbench detected."
    SetFont Amiga 8
ENDIF

IF EXISTS df0:devs/sysconf.norm
    Run sys:PopCLI 500 NEWCLI > NIL: CON:0/0/704/470/Terminal s:CLI.stu
ELSE
    Run sys:PopCLI 500 NEWCLI > NIL: CON:0/0/704/235/Terminal
    s:CLImin.stu
ENDIF
```

Hold it! What's this 704 x470 stuff. Isn't the maximum size 640 by 400?

Well yes, and no. The default maximum size for an interlace display is 640 pixels x 400 pixels. Another neat program in the Public Domain called MoreRows will allow you to expand the Workbench into the overscan area. Some monitors may have trouble with this and the windows will disappear off the edge of the display. I've pushed mine to the limit but you can set it to anywhere in between. MoreRows can be found on Fred Fish #54.

PopCLI is a nifty program that sits around waiting for you to press the Left Amiga key and the ESCape key at the same time. When this happens, a new CLI window will be opened with the values as defined above. You will find PopCLI on Fred Fish #40 and Amicus #15.

For those of you who have some programming background and have found the generic CLI environment stifling, don't give up hope. There are alternatives. In the next few issues we'll look at some programs that give us the kind of control that is so desperately needed.

If you have any questions or comments,  
please send them to:

**Rich Falconburg**  
c/o Amazing Computing  
P.O. Box 869  
Fall River, MA 02722

•AC•



# An Introduction to ARexx Programming

*Solve the classic Towers of Hanoi problem using ARexx*

by Steve Fajwiczewski

ARexx, introduced a while ago by Bill Hawes (of ConMan fame), provides Amiga users with an alternative programming language, as well as a superior script facility. ARexx also provides a means for separate programs to interact with each other, giving the appearance of total integration.

This article is not a review of the ARexx package; rather it is an introduction to the language, and a demonstration of some of its abilities.

To paraphrase the ARexx user manual, ARexx is a high level language particularly well suited as a command language. ARexx has a few unique features, some of which are discussed here.

## **Interpreted execution**

ARexx programs, like BASIC programs, are interpreted. This means that no time is wasted compiling and linking ARexx programs, but it also means that they will execute slower than equivalent programs in any compiled language. As with BASIC, source level debugging and tracing is readily available. However, an ARexx program does not have to be executed in a special environment the way an AmigaBASIC program does. If WShell is used, the program can easily be invoked by typing its name at the WShell prompt, otherwise the program is executed in a fashion similar to Amiga script file, with a command equivalent to EXECUTE (naturally, it is assumed that the ARexx server is already installed in the system).

## **Typeless data**

There are no types in ARexx; all variables are treated as character strings. This differs from most other languages that have varying degrees of data typing (BASIC has 3 types: integer, floating point, and strings). This means that the same variable may contain numeric data or character data at different times, so the validity of operations is dynamically checked during execution. For example, in the code segment

```
A = 123
B = A + 1
A = "cow"
B = A + 1
```

the fourth line will cause an execution error, because variable A contains non-numeric data at this point, and arithmetic operations are not valid on strings. The ARexx manual does not usually use the word "variable". Instead, "symbols" are discussed. There are four types of symbols:

1. "Fixed" symbols are BASICALLY numeric literals; they begin with a digit or a period.
2. "Simple" symbols are conventional variables. They do not contain any periods in them and they do not begin with a digit.
3. "Stem" symbols do not begin with a digit, and they have one period at the end of the name.
4. "Compound" symbols include one or more periods in the interior of the name.

## **Stems and Compound Variables**

ARexx allows you to create a collection of data fields known as a compound variable. A compound variable's name is divided into separate parts by periods. It has the structure

stem.n1.n2....nk

where the first part is called a stem, and parts n1 through nk (also called nodes) are either fixed or simple symbols. Whenever a compound variable is referenced, ARexx expands its name by substituting each node with its current value, then uses the expanded version instead of the compound variable.

Compound variables can be easily used to implement more conventional data structures found in other languages, such as arrays and records (structures, in C lingo). These may be thought of as dynamic arrays, where new elements may be added at runtime. This is very similar to Snobol tables.

Stem symbols provide access to a whole class of compound variables. For example, initializing a stem to 0 sets all compound variables that begin with the stem name to 0.

(continued)



# 3D OPTIONS

**CONVERT IFF BITMAPPED PICTURES TO  
2D AND 3D OBJECTS!!!**

**NOW YOU CAN MAKE USE OF ALL THOSE  
DPAINT AND DIGITIZED PICTURES IN  
YOUR FAVORITE 3D OR CAD PACKAGE.**

**VIDEOSCAPE 3D, AEGIS DRAW, MITOCAD, MCAD  
PROFESSIONAL PAGE, POSTSCRIPT  
DIF (AUTOCAD...) & MORE.**

**TO ORDER CALL  
1 800-628-2828  
EXT 829**

**WE ACCEPT**

**VISA, MASTER CARD AND AMERICAN EXPRESS**

**ONLY \$49.95**

**OR SEND CHECK OR MONEY ORDER TO:**

**RAINBOWS EDGE PRODUCTIONS  
4412 4TH AVE SUITE 2  
BROOKLYN N.Y. 11220**

**Also available: 3D MAPS, 3D FONTS, 3D UTILITIES (\$29.95)**

**All product names are Trademarks of their respective companies.**

## *Dynamic variable allocation*

As with BASIC, ARexx variables don't have to be declared before they are used. Unlike AmigaBASIC, however, variables are not automatically initialized to 0., they are initialized to their own name. For example, the statement

```
say j
```

will print "J". ARexx realizes that j was never assigned a value (and therefore it does not exist). Whenever j is referenced, ARexx substitutes the string "J" for its value.

## *Function Calls and Scoping of Variables*

ARexx supports function calls with parameter passing. Passing of parameters is done by value only (as opposed to by reference). In other words, parameters to a function can be used for input into the function, but not output from it. Functions can return a value however, just as in C and Pascal. Recursion (the act of a function calling itself) is also supported.

By default, a function uses the same "storage environment" as its caller. (A storage environment is the area in memory where all variables are allocated and maintained.) That means that if a function sets a variable to some value, that variable is affected on the level of the function's caller.

It is possible, though, to allocate a private storage environment for the function, by specifying the PROCEDURE statement after the function name. Doing this protects the caller's storage variable from any of the function's actions. If the function sets a variable to some value, that value will only be in effect while the function is running. The EXPOSE clause of the PROCEDURE statement allows a function to modify all variables mentioned in the EXPOSE variable list in the function caller's storage area. For example, in the program

```
/* */
VarOne = 1; VarTwo = 2; VarThree = 3
CALL FuncOne
SAY VarOne VarTwo VarThree
EXIT

FuncOne : PROCEDURE EXPOSE VarOne VarTwo
          VarOne = 15; VarTwo = 30; VarThree = 45
          SAY VarOne VarTwo VarThree

RETURN
```

The SAY statement within FuncOne will display "15 30 45". The same statement in the main program will display "15 30 3". VarThree was not affected by FuncOne because it was not exposed to it.

Please note that the method of referencing variables in ARexx functions is quite different from the method employed by languages like Pascal, Modula-2, C or PL/I. In those languages, when a variable is referenced in a function, the compiler first checks to see if that variable is local to the function (in ARexx terminology, if it exists in the function's private storage environment). If the variable is not local, the compiler will look for it in the calling function. If the compiler still cannot find the variable, it will repeat this lookup process until there are no more nesting of function code (i.e. the main program's storage environment was searched). This method of symbol lookup is termed "lexical scoping", as it uses the actual position of the function source code to expand the scope of the lookup.

ARexx, on the other hand, will first attempt to use the local storage environment. If it doesn't exist, the caller's environment is used for the symbol lookup. This lookup mechanism does not depend on the position of the function's source code, but rather on the sequence of execution. This is known as dynamic scoping.

Those who are experienced in Pascal, Modula-2, or C will have to keep this in mind, as it may make a difference. For example, in the program

```
/* */
CALL FuncOne
SAY FirstVar
EXIT

FuncOne : PROCEDURE EXPOSE FirstVar
          /* <- very important */

          CALL FuncTwo

RETURN

FuncTwo : PROCEDURE EXPOSE FirstVar
          FirstVar = 23

RETURN
```



93



```

call InitWindow(nrings)
signal on break_c /* listen to ^C from the user */
call InitObjects(nrings)
call InitialDisplay(nrings)

call hanoi0(nrings,1,2,3) /* Start movin' those rings! */
call GotoRC(stdout,HEIGHT+5,38)
say "Done!"
exit

/*****
/* Handle ^C from user */
*****/
BREAK_C:
call GotoRC(stdout,15,38)
say "Interrupted!"
exit

/*****
/* Display the initial state of */
/* the poles and rings. */
*****/
InitialDisplay: procedure expose FloorRow PoleSpacing Rings.
arg num
do i = 1 to num
call GotoRC(stdout,FloorRow - i + 1, PoleSpacing)
call CursorBack(stdout, Rings.i % 2)
call WriteCH(stdout,copies(' ',Rings.i))
end
return

/*****
/* hanoi0 - the main guy */
*****/
hanoi0: procedure expose Rings. Poles. PoleSpacing FloorRow ,
FlyRow nrings
arg num,From,To,Thru
if num = 1 then call MoveRing(From,To)
else do
call hanoi0(num-1,From,Thru,To)
call MoveRing(From,To)
call hanoi0(num-1,Thru,To,From)
end
return

/*****
/* Do the visual display of */
/* ring movement. */
*****/
MoveRing: procedure expose Rings. Poles. PoleSpacing FloorRow ,
FlyRow nrings
arg From,To

t = Poles.From.top
src = Poles.From.t
Size = Rings.src
HalfSize = Size % 2
StartCol = From * PoleSpacing - HalfSize
EndCol = To * PoleSpacing - HalfSize
RingStr = copies(' ',Size)
Blank = copies(' ',Size)
RestoredPoleString = overlay('|',Blank,HalfSize + 1,Size)
/* Pull ring of the source pole */
do i = (FloorRow - t + 1) to FlyRow + 1 by -1
call GotoRC(stdout,i, StartCol)
call WriteCH(stdout,RestoredPoleString)
call GotoRC(stdout,i-1, StartCol)
call WriteCH(stdout,RingStr)
end
incr = sign(EndCol - StartCol) * 2
/* Move ring over the destination pole */
do i = StartCol to EndCol - incr by incr
call CursorBack(stdout,size)
call WriteCH(stdout,Blank)
call CursorBack(stdout,size - incr)
call WriteCH(stdout,RingStr)
end
Poles.From.t = 0
Poles.From.Top = Poles.From.top - 1
Poles.To.top = Poles.To.Top + 1
dst = Poles.To.top
Poles.To.dst = src
/* Put ring on destination pole */
call CursorBack(stdout,size)
call WriteCH(stdout,Blank)
do i = FlyRow + 1 to (FloorRow - dst)
call GotoRC(stdout,i, EndCol)
call WriteCH(stdout,RestoredPoleString)
call GotoRC(stdout,i+1, EndCol)
call WriteCH(stdout,RingStr)
end
return

```

```

/*****
/* InitObjects - Set up data */
/* structures. */
*****/
InitObjects: procedure expose Rings. Poles.
arg num
j = 3
do i = num to 1 by -1
Poles.1.i = i
Poles.2.i = 0
Poles.3.i = 0
Rings.i = j
j = j + 2
end
Poles.1.top = num
Poles.2.top = 0
Poles.3.top = 0
return

/*****
/* InitWindow - clear screen */
/* and draw the poles. */
*****/
InitWindow: procedure expose FlyRow FloorRow PoleSpacing
arg num
call EraseScreen(stdout)
banner = overlay(num,"Towers of Hanoi: Moving x Rings",26)
call GotoRC(stdout,1,40 - (length(banner) % 2))
say banner
do n = 1 to 3
call GotoRC(stdout,FlyRow,n * PoleSpacing)
do i = FlyRow to FloorRow
call CursorDown(stdout,1)
call writech(stdout,'|')
call CursorBack(stdout,1)
end i
end n
return

/*****
/* Cursor movement routines */
*****/

/*****
/* Move cursor to row/col */
*****/
GotoRC:
arg out, row,col
call writech out, '9B'x || row || ';' || col || 'H'
return

/*****
/* Move cursor down */
*****/
CursorDown:
arg out,num
call writech out, '9B'x || num || 'B'
return

/*****
/* Move cursor to the left */
*****/
CursorBack:
arg out,num
call writech out, '9B'x || num || 'D'
return

/*****
/* Move cursor to up */
*****/
CursorUp:
arg out,num
call writech out, '9B'x || num || 'A'
return

/*****
/* Move cursor to top left and */
/* erase screen */
*****/
EraseScreen:
arg out
call GotoRC(out,1,1)
call writech out, '9B'x || '2J'
return

```



# Crunchy Frog

## *Amiga C Programming for Programmers*

by Jim Fiore

I know what you're thinking: "Crunchy Frog? What's that? And what does it have to do with programming an Amiga?" Well, it has nothing to do with programming, but it got your attention. Besides, I like the way it sounds (so does Monty Python). The purpose of this series is to teach programmers how to write Amiga applications in the C language. This assumes that you already have a working knowledge of C, although you certainly don't have to be an expert. We're going to be covering a host of items that will enable you to utilize screens, windows, requesters, slide controls, string gadgets and the like.

Generally speaking, C is a good language to choose if you're doing serious Amiga development work. You will find that nearly all the documentation is in C. Of course, C-generated programs are neither as small nor as fast as good assembler programs, but you can always include assembly sections for speed critical areas.

First of all, the two items we are utilizing are Amiga Exec (the multi-tasking heart of the machine), and Intuition, which serves as a mouse/icon oriented 'front end' to your application. If you learned C on a typical command line oriented system, you are going to have to unlearn a few things in order to make Intuition-based programs. For example, you will generally not be using functions like `printf()` and `fopen()`. Also, you'll encounter a number of new items if you've never worked with a multi-tasker before.

Don't fret though. The jump to light speed is really quite straightforward. You may find certain resources to be invaluable. I strongly recommend the Amiga Programmer's Handbook, by Eugene Mortimore.

Usually, the one thing that confuses people about a given programming environment is not so much the task at hand, but the terminology and jargon. You're going to be hearing the following terms quite a bit, so here's a brief description of each item.

**Screen:** A work area that can usually be depth arranged and slid up or down with the mouse. Screens have specific graphic attributes including mode and number of available colors.

**Window:** A work area associated with a given screen. Windows can usually be depth arranged, moved about the enclosing screen, resized and the like. Several windows may be opened inside a given screen. Each window inherits the graphic attributes of its parent screen.

**Menu:** Menus can be either text or image-based. They are associated with windows and are one form of obtaining input from the user.

**Gadget:** This is the other major input form. Gadgets are broken into two broad classifications: Those that you don't fiddle with (i.e., System gadgets like the depth arranger or resizer), and those that you generate and monitor. There are three basic types: String, for text input, Boolean, for yes/no choices (like an on/off button), and Proportional, for value input (like RGB color sliders). Gadgets may be associated with Windows, Screens or Requesters.

**Requester:** A special purpose work area, usually containing a collection of gadgets - for example, a file requester. Requesters cannot be moved.

**IDCMP:** Intuition Direct Communications Message Port. This is how your application learns of outside events, such as mouse movements or menu choices.

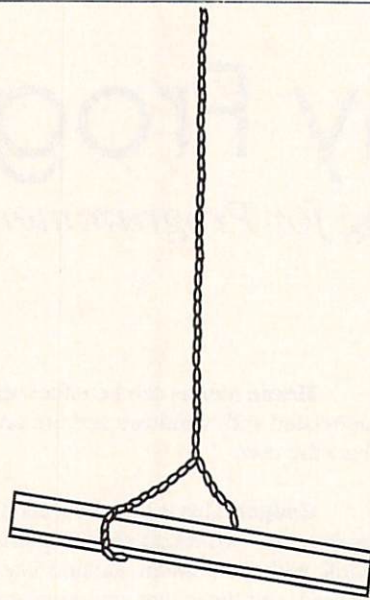
**Library:** To save on duplication of code in a multi-tasking OS, common functions are collected into libraries. A library must be opened before its functions are called, and it must be closed before the program exits.

**Device:** Access to hardware is normally done through device drivers. This presents a reasonably consistent programmer's interface and it allows for multi-tasking. (It is possible to go directly to the hardware for more demanding applications, although this is frowned upon in some circles.)

One of your first jobs in `main()` will be to open up the libraries that will be needed. Once this is done, you will need to open a window (possibly windows), and usually a screen as well. All rendering - whether graphics or text - is done in windows. Usually, windows will have menus and gadgets attached to them. (It is through the window's message port that you will normally learn of user events). In essence, your program will be running an infinite loop - waiting for, then responding to user signals until termination. This loop, sometimes referred to as an IDCMP loop, or event processing loop, is NOT a busy-wait loop. There is a special 'go to sleep until it's time to do something' function, called `Wait()`. By using `Wait()`, the program will not waste CPU time.

(continued)





**Under Construction**

**Watch for something  
new and exciting  
on March 3rd**

## Amazing Advertisers Index

### Discover something interesting?

### Need more information?

Use the Reader Service Card in every issue of AC to contact those advertisers who have sparked your interest. Advertisers want to hear from you. This is the best way they have of determining the Amiga community's interests and needs. Take a moment and contact the companies with products you want to know more about. And, if you do contact an Amazing Advertiser directly, please tell them you saw them in AC

Advertiser	Page	Reader Service Number
Advanced Computer Services West	100	200
Amazing Computer Systems	44	144
AMI Expo	41	141
Antic Publishing	55	155
Capilano Computing	89	189
Creative Solutions	22	122
D-Five Associates	88	188
Delphi Noetic Systems	104	204
E Z-Soft	97	197
Elipsys	99	199
Emerald Intelligence	9	109
Erich Stein & Associates, Inc.	64	164
Expansion Technologies	19	119
Fishers Computers and Software	80	180
Flexible Data Systems	39	139
Gramma Software	80	181
Great Valley Products	7	107
Great Valley Products (Dealers Only)	7	108
Interactive Video Systems	61	161
Lattice	11	111
Megatronics	CII	252
Micro Masters	12	112
Micro Systems Software	17	117
Micro Way	33	133
Microbotics	13	113
Microsmiths	46	146
New Horizons Software	25	125
New Tek	CIV	254
New Wave Software	27	127
OTG Software	86	186
Oxxi, Inc.	1	101
Practical Solutions	75	175
Quality Business Systems	80	182
Ronin Research and Development	4	104
Rainbows Edge Production	92	192
Sedona Software	21	121
Soft Link	69	169
Software Advantage Consulting Corp.	54	154
Software Sensation	60	160
Software Terminal	29	129
The American Cancer Society	30	130
The Bit Bucket Computer Store	76	176
The Memory Location	45	145
The Trumor Company	78	178



The one thing you may notice about Intuition programming is that there are a number of different structures floating around - structures for Screens, Windows, Requesters, Menus, etc. In many cases, structures will be initialized at the head of the program or module. Our example program, CrunchyFrog1, opens a screen and one window. A sine wave will be plotted in the upper section of the window and mouse coordinates will be printed in the lower section. This will show you how to open the appropriate libraries, create a screen and window, define a color scheme, draw arbitrary graphics and text into a window, and scan the IDCMP.

CrunchyFrog1 also contains a large amount of code (like dummy function calls) that we will use in the future. Generally speaking, all simple numeric defines are shown in UPPER\_CASE. All Exec/Intuition function calls (and simple function defines) use upper case first letters, as in AnExecFunction(). Our own functions and variables will be lowercase, as in our\_function(). I do all of my development work with the Manx compiler. There may be some minor changes for Lattice users.

First off, we have to include the right header files. A close look at something like Intuition.h indicates that it has its own include list. We need math.h since we will be using the trig function sin(). After this come the defines. To print text onto a window, we must first Move() to the desired location, and then print using the Text() function. I've combined these two items into the simple define MyText().

You will notice a reference to a 'RastPort'. Basically, a RastPort is a drawing area. All graphics functions utilize RastPorts. Each Window, Screen and Requester has its own RastPort, and that's how you distinguish between different drawing areas.

Next come our globals. First we have the Intuition and Graphics (Gfx) library bases, followed by window, screen, and a viewport pointer for the screen. We will use the ViewPort as a means of changing the color palette. It is very important that these items be initialized to 0 for our cleanup routine to work properly (I have seen many examples that do not do this!). The NewScreen and NewWindow structures follow. Do not confuse these with the Screen and Window structures. NewScreen and NewWindow are only needed to set up a screen or window, and are not used after that.

Our screen will be high res, non-interlaced with 3 bitplanes (8 colors max). The screen size is 640 x 200 pixels. This is a custom (non-Workbench) type, and we wish to create the screen behind any other screens and then bring it to the front when creation is complete. (It's a little prettier.) We have also specified that the screen's font will be the 80 char topaz type (note the Text Attribute structure definition).

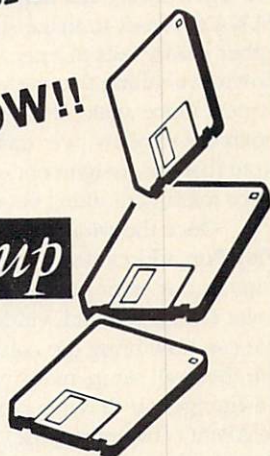
Our window will start at 250 x 120 (alterable from 100 x 50 to 640 x 200), and will be offset from the screen by 50 x 20 initially. By the way, all offsets are relative to the top left corners (this applies to things like borders, images, etc. as well). Our detail and block drawing pens are set to the defaults (-1).

The window will have four system gadgets ready to accept input: close, drag, size, and depth arrangement. It will become the active window as soon as it opens. It will also send mouse information and be of smart refresh type (you don't have to worry about redrawing the window contents if another window is moved over it). Finally, we define our own color

BREAKTHROUGH

**All  
Hard Drive Back-Up  
Programs were created  
equal . . .  
UNTIL NOW!!**

**EZ-Backup**



**100% better than any other hard drive back-up program  
EZ-Backup is a genuine breakthrough**

**EZ-Backup actually manages  
the space on your backup disks--**

Before EZ-Backup the only way to get rid of the piles of incremental back-up disks that littered your desk was to do another full back up. This method is inconvenient and requires a whole new set of unformatted floppy disks.

With EZ-Backup you use the same set of disks for every incremental back-up. Only one full back-up required--Ever! Space on the disks is managed by deleting obsolete archive files and allowing you to save from 0-255 versions of each file. **Your files are saved--even if you have completely deleted them from the hard drive!**

**EZ-Backup comes with an optional warning screen--**

We all tend to put off doing backups. EZ-Backup's warning screen reminds you. If you would rather not be reminded--you have the option to shut the warning screen off.

**EZ-Backup prevents you from damaging valuable data--**

By checking the volume label, EZ-Backup keeps you from writing over important files.

**EZ-Backup uses Standard Amiga format--**

Files are archived in standard Amiga format and work with all standard utilities.

**EZ-Backup provides easy recovery of individual files--**

A simple to use mouse-oriented program allows you to recover individual files.

**Works with all Amiga-DOS compatible hard drives.  
(Amiga-DOS version 1.2 or higher)**

- |   |                                 |
|---|---------------------------------|
| * Provides archive-bit utilities  | * Not copy protected            |
| * Multi-tasking   | * Complete manual with examples |
| * Upgrades provided free for the first six months after program purchase! |                                 |
| * Free telephone support!   |                                 |

**Not more expensive just the best - \$49.95 - from:  
EZ-SOFT or an Amiga Dealer near you.**

21125 Chatsworth Street  
Chatsworth, CA 91311  
(818) 341-8681

**Dealer Inquiries Welcome**

**EZ-SOFT**

(continued)



table with eight elements. The three digits each represent a red, green, or blue value from 0-f (in hex). For example, 0xf00 is pure red, while 0x0f5 is full green with a small amount of blue.

In main(), we first call open\_all() which opens our Intuition and Gfx libraries. OpenLibrary() returns a pointer. If the OpenLibrary call fails, it will return 0. You should ALWAYS check to make sure that calls like this succeed! If either library fails to open, we call damp\_mop(), which closes down everything that we've opened. After this, we open our screen. Once again, if the pointer equals 0, we exit. Before we open the window, we must indicate what screen it belongs to. Note that this assignment cannot be done at the initialization, since main\_scrn didn't yet exist.

Once the window is opened (main\_wind), we obtain the ViewPort address for use in the LoadRGB4 function. This will copy in our color palette in place of the default palette. At this point our screen and window are open, and the colors are set. We can now bring the color palette to the front of the display. We then call setup\_main\_wind(), which draws the sine wave. To change our drawing pen colors, we use the function SetAPen(). To draw a line, Move() to the starting point and then Draw() to the ending point. Note that the sine wave is actually a series of very small line segments, since it does repeated calls to Draw(). (Draw() does an implied Move(), so Move() is not required each time.) It is very important that you don't attempt to stuff floating point values into the Move() and Draw() functions! Finally, we make a call to our MyText macro to print the letters 'X' and 'Y'.

Back in main() we enter the IDCMP loop. We must put our task to sleep while it waits for the user's message. The Wait() function takes as its argument a signal bit number (it may take a number of them ORed together). We can derive the signal bit from the UserPort field of the Window structure. The value of mp\_SigBit tells the bit POSITION. A one is then shifted left to obtain the wait bit. Once the task is signaled, it wakes up. Next, we test to see which signal woke up the task. (We only have one right now, so this is a tad overblown.) If the message originated from main\_wind, we would go off to handle\_main\_messages().

Once in handle\_main\_messages(), we set up a while loop to read main\_wind's message port. It is quite likely that we may have more than one message waiting for us. The IntuiMessage structure contains a number of fields, each with pertinent information. We obtain the next message in the queue with the GetMsg() function. We only have temporary license to use this data, so we make local copies of the data and then ReplyMsg(). You should make sure that you Reply to all messages. If you don't, your program will eventually crash.

At this point we can determine the basic type of message by looking at the Class field. The possibilities are the same as those set back in our NewWindow structure. Virtually all of the calls here are to dummy functions which we'll use later, except for CLOSEWINDOW which calls damp\_mop() for an orderly shut down, and MOUSEMOVE which sets the mouse\_moved flag. Once the while loop exits, if there has been a mouse move, the update\_coords() function is called to print the new mouse position. This is done because it is less CPU intensive than constant updates. Once the function exits, our program calls Wait() and goes back to sleep.

The output of the program is shown in figure one. Once the sine wave is drawn, you can move the mouse around the screen and watch the coordinates change. There are two things you should notice about the coordinates. First, the reported values are always relative to the window's upper left corner no matter where in the screen the window is placed. If you move off to the left or above the title bar, you will get negative coordinates. Second, you will only update the values when the window is active. If you click on the screen, you will notice that the window is deactivated (ghosted title bar) and updating stops. You can resize the window, but notice that if you shrink the window, and thereby destroy the sine wave, then resizing the window larger will NOT recreate the graph. This can be remedied by using a super bitmap window, or by calling setup\_main\_wind() when appropriate NEWSIZE events are received. Note that coordinates ARE reprinted since the update\_coords() function is repeatedly called.

At this point I suggest that you experiment with the program by altering functions like setup\_main\_wind() with different pen numbers and commands. You may also wish to change the system color table and a few of the system window gadgets (but DON'T remove CLOSEWINDOW or WINDOWCLOSE, as you'll have no way of exiting the program!) Another interesting item is to delete the SCREENBEHIND flag to watch everything being built in sequence. (It's pretty quick.) In any case, have fun, and we'll continue CrunchyFrog next time.

### Author's info

Jim Fiore is the resident C programmer at dissidents in Utica, NY. He has a number of years teaching experience in Electrical Engineering Technology. He may be contacted through BIX as jfiore.

```
/*
CrunchyFrog1.c

Jim Fiore @ dissidents 7/9/88. This program is copyrighted, however,
you can use it, abuse it, or do just about anything you'd like with
it ( between consenting adults and/or silicon units ), with the
exception of distribution for a profit.

compiled and linked with Manx Aztec C v3.6 under AmigaDOS 1.2

cc +L CrunchyFrog1.c

ln CrunchyFrog1.o -lm132 -lc32

~/

#include "functions.h"
#include "intuition/intuition.h"
#include "math.h"

/* ----- defines ----- */

/* MyText(*RastPort, Xposition, Yposition, char *buffer) */

#define MyText(r,x,y,b) Move((r),(x),(y)); Text((r),(b),strlen(b))

#define INTUITION_REV 33L
#define GRAPHICS_REV 33L
#define DEPTH 3
#define MAX_COLORS 8 /* 2 raised to DEPTH */

/* ----- Globals ----- */
```



```

struct IntuitionBase *IntuitionBase=0L;
struct GfxBase *GfxBase=0L;
struct Window *main_wind=0L;
struct Screen *main_scrn=0L;
struct ViewPort *view_port=0L;

struct Menu main_menu[]={NULL}; /* to be used in the future */

/* use the 80 character topaz font for the screen lettering */

struct TextAttr topaz80_font_attr={
    (UBYTE *)"topaz.font",
    TOPAZ_EIGHTY,
    FS_NORMAL,
    FPE_ROMFONT };

struct NewScreen ns={
    0, 0, /* LeftEdge, TopEdge */
    640, 200, /* Width, Height */
    DEPTH, 0, 1, /* Depth, DetailPen, BlockPen */
    HIRES, /* ViewModes */
    SCREENBEHIND | CUSTOMSCREEN, /* Type */
    &topaz80_font_attr, /* Font */
    (UBYTE *)"Dinsdale's Screen ", /* DefaultTitle */
    NULL, NULL }; /* Gadgets, CustomBitMap */

struct NewWindow m_nw={
    50, 20, /* LeftEdge, TopEdge */
    250, 120, /* Width, Height */
    -1, -1, /* DetailPen, BlockPen (default) */
    /*
    MENUPIK | GADGETUP | GADGETDOWN | \
    RAWKEY | CLOSEWINDOW | MOUSEMOVE | \
    MOUSEBUTTONS | NEWSIZE, /* IDCMPFlags */
    SMART_REFRESH | ACTIVATE | \
    WINDOWresizing | WINDOWCLOSE | \
    WINDOWDEPTH | WINDOWDRAG | \
    REPORTMOUSE, /* Flags */
    NULL, NULL, /* FirstGadget, CheckMark */
    (UBYTE *)" Main Window ", /* Title */
    NULL, NULL, /* Screen, BitMap */
    100, 50, /* MinWidth, MinHeight */
    640, 200, /* MaxWidth, MaxHeight */
    CUSTOMSCREEN }; /* Type */

USHORT sys_color_table[MAX_COLORS]={ 0xabc, 0x130, 0xf00, 0xf40, 0xb20,
    0x34f, 0xb0e, 0x48b};

VOID open_all(), damp_mop(), handle_main_messages(), setup_main_wind(),
\
    handle_main_rawkey(), handle_main_menu(), handle_main_gadgetdown(),
\
    handle_main_gadgetup(), handle_main_mousebuttons(),
handle_main_newsize();

/*-----start of main()-----*/

main()
{
    LONG main_wait_bit, wait_mask;

    /* -- open Intuition and Graphics libs -- */

    open_all();

    if ( (main_scrn = (struct Screen *)OpenScreen(&ns)) == NULL )
        damp_mop();

```

(continued)

## Would you use your Amiga like this?



### Without ENCORE you might as well be!

Encore allows you to build macros by recording mouse movements and keystrokes. You can assign each macro to a key combination so that with a touch of the finger any macro can instantly be played back.

Macros can have spoken narration.

Macros can have scrolling text at the bottom of the monitor.

Macros can loop back for repeated playback

Macros can be either relative or absolute.

Macros can be appended.

Macros can be user interactive.

Macros can be time or file compressed.

Macros can be of unlimited size.

Automatic mouse and keyboard lockout.

Works with any Amiga program.

Speech preferences program with phonemized word database.

**No matter what you are using your Amiga for, Encore is a must for your software library. See your local dealer for details.**

ENCORE is NOT COPY PROTECTED Suggested retail \$69.95  
Dealer inquiries invited.

**ELIPSYS**

PO Box 16757  
Seattle, Wa 98116

(206)938-0825  
FAX (206)932-9520



# ADVANCED COMPUTER SERVICES WEST

•PRESENTS•

## "THE AMIGA SERVICE CLUB"

### RECEIVE-

- Newsletters
- Reviews
- B.B.S.
- Free Access To Public Domain Library
- Wholesale Buying Club

### ENROLL NOW!

\$20. Membership Fee (1 year)

Club Membership Is Limited To 500

ADVANCED  
COMPUTER SERVICES WEST  
P.O. BOX 1599  
LOMITA, CA 90717

•Amiga is a trademark of Commodore-Amiga

```
m_nw.Screen=main_scrn;

if ( (main_wind = (struct Window *)OpenWindow(&m_nw)) == NULL)
damp_mop();

/* set screen colors to our choices */
view_port = ViewPortAddress(main_wind);
LoadRGB4(view_port, sys_color_table, MAX_COLORS);
ScreenToFront(main_scrn);

/* do some drawing */
setup_main_wind();

/* - set up IDCMP read loop - */

/* This is rather round about for a single window as you can just say
Wait( 1<<main_wind->UserPort->mp_SigBit );
handle_main_messages();

We're going to look at multiple windows in the future, and the
version
below will be easier to expand */

FOREVER /* also known as for(;;) */
{
    main_wait_bit = 1<<main_wind->UserPort->mp_SigBit;
    /* get main_wind's
    signal bit */
```

```
wait_mask = Wait( main_wait_bit );
/* go to sleep til user does something */

if (wait_mask & main_wait_bit) /* true if main_wind woke up */
    handle_main_messages();

}

/* end of main() */

/*----- opens Intuition and Graphics libs -----*/
VOID open_all()
{
    IntuitionBase=(struct IntuitionBase *)OpenLibrary("intuition.library",
    INTUITION_REV);

    if (IntuitionBase==NULL) damp_mop();

    GfxBase=(struct GfxBase *)OpenLibrary("graphics.library",
    GRAPHICS_REV);

    if (GfxBase==NULL) damp_mop();
}

/*----- closes windows, screen, Graphics, Intuition -----*/
VOID damp_mop()
{
    struct IntuiMessage *mes;

    if ( main_wind )
    {
        /* Drain the IDCMP. Actually, this isn't really required as this
        memory will be reclaimed, but I sleep better if I do it */
        while( mes=(struct IntuiMessage *)GetMsg( main_wind->UserPort ) )
            ReplyMsg( mes );

        if ( &main_menu[0] ) ClearMenuStrip( main_wind );
        CloseWindow( main_wind );
    }

    if ( main_scrn ) CloseScreen( main_scrn );
    if ( GfxBase ) CloseLibrary( GfxBase );
    if ( IntuitionBase ) CloseLibrary( IntuitionBase );
    exit( FALSE );
}

VOID setup_main_wind() /* draws x,y axis and a sine wave */
{
    struct RastPort *rast = main_wind->RPort;
    double x, y;

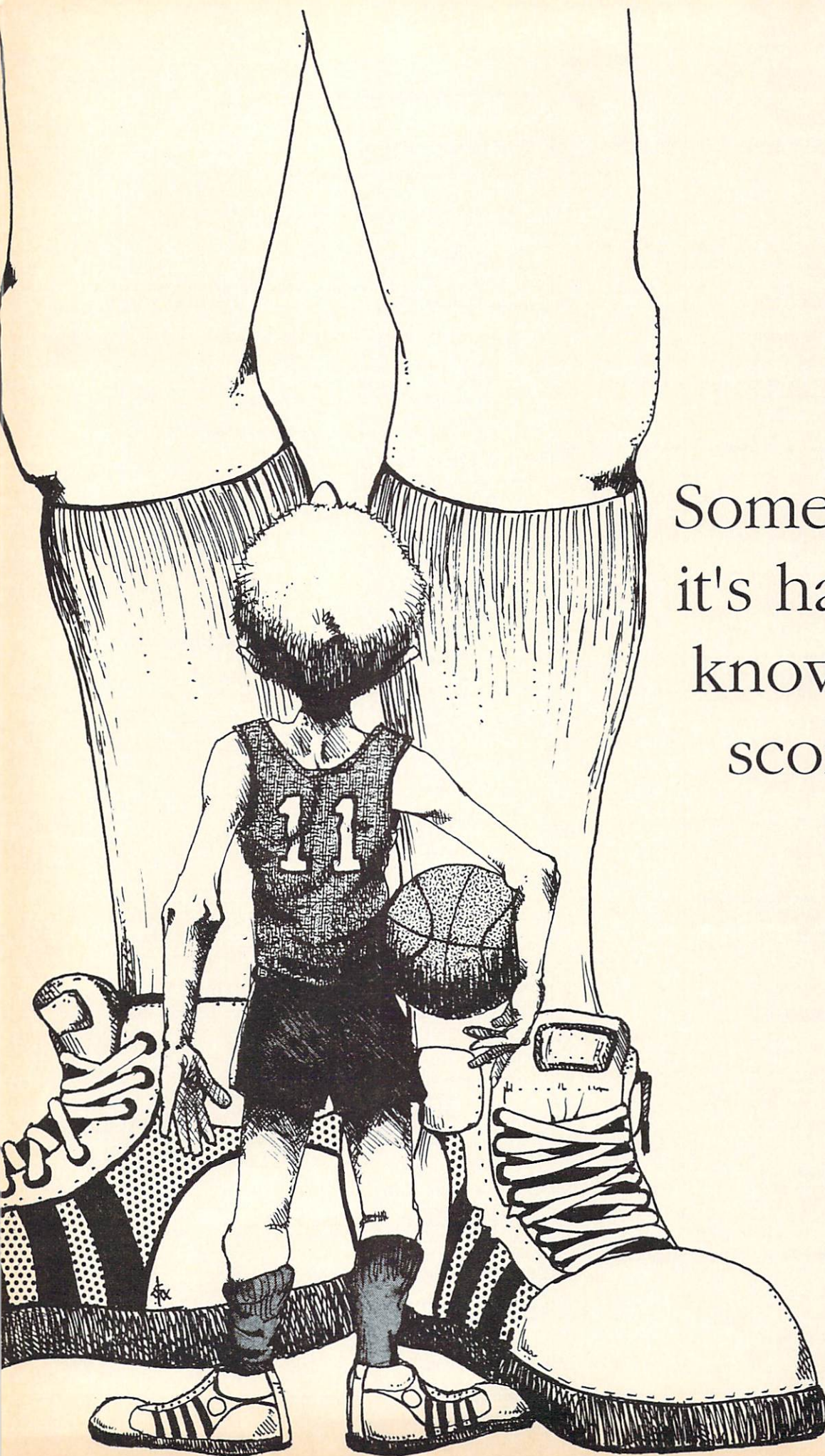
    /* Draw x, y axis. First set the pen color to pen 4. The origin will be
    at point 20, 50, the max swing will be +/- 35 pixels, and the length
    will be 200 pixels */

    SetAPen( rast, 4);

    Move( rast, 20, 15);
    Draw( rast, 20, 85);
```

(continued)





Sometimes  
it's hard to  
know the  
score...



```

Move( rast, 20, 50);
Draw( rast, 220, 50);

/* Now for the sine wave. Move to the origin, and to see it more
clearly,
change to pen 5 */

Move( rast, 20, 50);
SetAPen( rast, 5);

for( x=1.0; x<200.0; x=x+1.0)
{
    y = -1.0 * (35.0 * sin( x/10.0 ));

    /* Offset y so it straddles the x axis */
    y = y + 50.0;

    /* Plot line segment. Note the x offset. Cast is important! */
    Draw( rast, (SHORT)x+20, (SHORT)y );
}

/* Print the mouse position titles (X, Y) next to where the values will
appear */

SetAPen( rast, 6);
MyText( rast, 20, 100, "X");
MyText( rast, 80, 100, "Y");

}

VOID update_coords( x, y ) /* prints the mouse x y position */
SHORT x,y;
{
    UBYTE buf[5];
    struct RastPort *rast = main_wind->RPort;

    sprintf( buf, "%4d", x);
    MyText( rast, 30, 100, buf);

    sprintf( buf, "%4d", y);
    MyText( rast, 90, 100, buf);
}

/* dummy handlers, to be used in the future, follow */

VOID handle_main_mousebuttons( code )
USHORT code;
{
}

VOID handle_main_rawkey( code, qualifier )
USHORT code, qualifier;
{
}

VOID handle_main_menu( code )
USHORT code;
{
}

VOID handle_main_gadgetdown( address )
APTR address;
{
}

VOID handle_main_gadgetup( address )
APTR address;
{
}

VOID handle_main_newsize( window_ptr )
struct Window *window_ptr;
{
}

```

```

/*----- IDCMP routine -----*/

VOID handle_main_messages()
{
    struct IntuiMessage *message;
    SHORT mx, my;
    static SHORT mouse_moved;

    mouse_moved = FALSE;

    /* As long as we have messages in the que, make local copies of
appropriate data, reply, and then process accordingly */

    while( message=(struct IntuiMessage *)GetMsg( main_wind->UserPort ) )
    {
        ULONG class = message->Class;
        USHORT code = message->Code;
        USHORT qualifier = message->Qualifier;
        APTR address = message->IAddress;
        struct Window window_ptr = message->IDCMPWindow;

        mx = message->MouseX;
        /* Declared above. Must be local to entire func */
        my = message->MouseY;
        /* if we use the mouse move collection technique */

        ReplyMsg( message );
        switch( class )
        {
            case MOUSEMOVE:
                mouse_moved = TRUE;
                /* update_coords( mx, my ); */
                /* <-uncomment this for 'real time' updates */
                break;

            case MOUSEBUTTONS:
                handle_main_mousebuttons( code );
                break;

            case CLOSEWINDOW:
                damp_mop();
                break;

            case RAWKEY:
                handle_main_rawkey( code, qualifier );
                break;

            case MENUPICK:
                handle_main_menu( code );
                break;

            case GADGETDOWN:
                handle_main_gadgetdown( address );
                break;

            case GADGETUP:
                handle_main_gadgetup( address );
                break;

            case NEWSIZE:
                handle_main_newsize( window_ptr );
                break;

            /* we could add other choices here as well */
        }

        /* end of while(message..) */

        if ( mouse_moved ) update_coords( mx, my );

    } /* end of handle_main_messages() */

    /****** Dat's all folks....
    *****/
}

```



...until you check the "Board"

**Amazing**  
COMPUTING™

3	5	7	4*
---	---	---	----

Pages Published

6	8	6*
---	---	----

Articles Published

From our very first issue, Amazing Computing has always delivered more. Each issue of AC delivers more information, more completely presented on more topics for the Amiga community than any other resource. Amazing Computing has chronicled the growth of the Commodore Amiga from Abasic to ARexx by providing the most complete and comprehensive publication available.

**Over 50% more Amiga information!**

Amazing has produced over 17% more pages dedicated to the Amiga than its nearest competitor. AC has produced over 50% more articles for the Amiga. **That's over 50% more Amiga information!**

Not only is Amazing Computing the most prolific Amiga resource available, but AC covers the

subjects you want to know. AC's thorough coverage through special features and indepth articles of Amiga interests in Video, Databases, Programming, Entertainment, Music, Animation, and more is legendary.

And, best of all, Amazing Computing is written by Amiga users. AC does not have a group of staff writers producing stale Amiga coverage. AC articles are written by Amiga users all over the world. Real Amiga users who want to explore more of the Amiga because it is fun. This excitement makes AC the #1 Amiga resource.

Join the Amiga excitement. Subscribe to the #1 resource for the Commodore Amiga. Subscribe to Amazing Computing and don't miss a critical point of Amiga interest again.

\* Figures based on AC January 1989 V4.1



# The **F-BASIC™** Language System

The **FASTEST** Growing  
**FASTEST** Performing  
Development Environment  
For The Amiga!

A Beginner Can  
Immediately Use  
**F-BASIC™!**

Available Soon...VERSION 2.0 of F-BASIC  
New Features: IFF & Random Access File Support, High  
Level Animation, Double Precision, Icon Support, & F-BASIC Linker  
SAVE! Buy Now & Upgrade to 2.0 Upon Availability For Less Money!

An  
Expert Can  
Never Outgrow  
**F-BASIC™!**

## The **F-BASIC™** System An Enhanced Compiled BASIC Language

Compare These Features:

- Extensive Control Structures
- LOCAL & GLOBAL Variables
- Ultra Fast Floating Point
- RECORD Structures & Pointers
- INCLUDE & APPEND Separate Files
- PATTERN Matching Support
- Easy high level access to AMIGA Screens, Windows, Menus, Sound, Speech, Graphics and Events.
- Recursive SUBROUTINES & FUNCTIONS
- 32, 16, and 8 Bit INTEGERS
- Powerful String Facilities
- Direct ROM Kernel Access
- Bitwise Operations
- Access to 68000 Registers
- Sample Programs Disk & User's Manual

### The **F-BASIC™** System Also Has A Source Level Debugger

- Debug F-Basic programs at the SOURCE level.
- Fully windowed Intuition interface—windows for Source Code, 68000 Registers, Memory Dumps, Program Variables, etc.
- Set break points and single step trace
- Display all variables, arrays, or RECORDS by name.
- Full reverse-assembler included.
- A comprehensive User's Manual with full documentation of all SLDB supported features and windows.

The **F-BASIC™** Language System (1.0)—Only \$79.95  
**F-BASIC™** System and Complete SLDB—Only \$129.95

SEND CHECK OR MONEY ORDER TO:

**DELPHI NOETIC SYSTEMS, INC.**

Post Office Box 7722  
Rapid City, South Dakota 57709-7722  
Credit Card or C.O.D. Call (605) 348-0791

F-BASIC AND FastCom are registered trademarks of DNS, Inc.  
AMIGA is a registered trademark of Commodore/AMIGA Inc.

(C Notes, continued from page 78)

```
item.elec.value = 2350.00;
list_item(&item); /* call the printing function */

setmem(&item, sizeof(item), 0); /* Clear structure */
item.type = ELECTRONIC; /* Electronics */
strcpy(item.elec.name, "CD Player");
item.elec.serial = 987654;
item.elec.value = 295.00;
list_item(&item); /* call the printing function */

} /* end of program and function main */

/* this functions accepts a pointer to an ITEM structure and after */
/* determining what the type is, will print the description */

list_item(item)
union ITEM *item; /* item passed */
{
    if (item->type == FURNITURE) { /* furniture detected */
        printf("Name: %s Color: %s Value %8.2f\n", item->furn.name,
            item->furn.color, item->furn.value);
    }
    if (item->type == ELECTRONIC) { /* electronic detected */
        printf("Name: %s Serial: %ld Value %8.2f\n", item->elec.name,
            item->elec.serial, item->elec.value);
    }
}
```

After the include files you will see that I have two definitions that are used in the program called FURNITURE and ELECTRONICS. These are not variables, rather, they are "equates". The compiler substitutes the number found after the definition in every location where the word occurs. Although in this example it would have been easier to type 1 or 2 instead of a long word, two advantages are achieved using #defines (pronounced "pound" defines). First, they make the code more readable, and second, if one of the values has to be changed, you would only have to change one location and recompile.

Next you will see that I have defined two structures. The FURN is for furniture variables and ELEC is for electronics. Although these structures could be rearranged to appear more similar, this demonstrates how union members can be different. The furniture structure contains a type, name, color, and value. Electronics contain a type, name, serial number, and value.

Notice that inside the ITEM union not only are the two types of structures identified, a single type variable is also defined. This lets us decide the variable type without having to use one of the structure names. We don't really need the addition type defined, but it does not cost us anything since the structures are going to be larger anyways. (Remember, the amount of storage will be the size of the largest item in the union.)

As you can see, this program assigns different types of furniture and electronics into the variable and then sends it off to be printed by the list\_item function. The list\_item function decides what type of item was received and then prints it using the appropriate method. If you notice the %8.2f contained in both printf function calls, that indicates the format to use to print the floating value. Finally, notice the first statement in each of the series of statements that assigns values into the item union. The function setmem is being used to fill the variable with zeros. This assures that no "extra" characters are floating around in the variable from a previous statements.

If my attempt at explaining unions has left you with a question or two, get out your reference materials and read what they say about unions. Also, don't be afraid to ask fellow programmers and, by all means, try a few experiments.

•AC•







<p><b>Texts:</b></p> <p><b>FractKeys</b> explains how to read function keys from Amiga Basic</p> <p><b>HackerSin</b> explains how to win the game "hacker" by installing a 68010 in your Amiga</p> <p><b>PrinterTip</b> sends escape sequences to your printer tips on setting up your startup-sequence file list of Transformer programs that work</p> <p><b>Printer Drivers:</b></p> <p>Printer drivers for the Canon PJ-1060A, the C-100 Prowler, an improved Epson driver that eliminates streaking, the Epson LQ-800, the Gemini Star-10, the NEC 8025A, the Okidata ML-92, the Panasonic KX-P10xx family, and the Smith-Corona D300, with a document describing the installation process.</p> <p><b>AMICUS Disk 10</b> <b>Instrument sound demos</b></p> <p>This is an icon-driven demo, circulated to many dealers. It includes the sounds of an acoustic guitar, an alarm, a banjo, a bass guitar, a bolin, a callopie, a car horn, claves, water drip, electric guitar, a flute, a harp arpeggio, a kickdrum, a marimba, a organ minor chord, people talking, pigs, a pipe organ, a Rhodes piano, a saxophone, a sitar, a snare drum, a steel drum, bells, a vibraphone, a violin, a wailing guitar, a horse whinny, and a whistle.</p> <p><b>AMICUS Disk 11</b> <b>C programs</b></p> <p><b>drutil</b> Intuition-based, CLI replacement manager</p> <p><b>cpri</b> shows and adjusts priority of CLI processes, S-E</p> <p><b>ps</b> shows info on CLI processes, S-E</p> <p><b>vidtex</b> displays CompuServe RLE pics, S-E</p> <p><b>AmigaBasic programs</b></p> <p><b>pointered</b> pointer and sprite editor program</p> <p><b>optimize</b> optimization example from AC article</p> <p><b>calendar</b> large, animated calendar, diary and date book program</p> <p><b>amortize</b> loan amortizations</p> <p><b>brush2BOB</b> converts small IFF brushes to AmigaBasic BOB OBJECTS</p> <p><b>grids</b> draw and play waveforms</p> <p><b>hilbert</b> draws Hilbert curves</p> <p><b>madlib</b> mad lib story generator</p> <p><b>mailtalk</b> talking mailing list program</p> <p><b>meadows3D</b> 3D graphics program, from A C™ article</p> <p><b>mousetrack</b> mouse tracking example in hires mode</p> <p><b>slot</b> slot machine game</p> <p><b>stactoe</b> the game</p> <p><b>switch</b> pachinko-like game</p> <p><b>weird</b> makes strange sounds</p> <p><b>Executable programs</b></p> <p><b>cp</b> unix-like copy command, E</p> <p><b>cls</b> screen clear, S-E</p> <p><b>diff</b> unix-like stream editor uses 'diff' output to fix files</p> <p><b>pm</b> chart recorder performances indicator</p> <p><b>Assembler programs</b></p> <p><b>cls</b> screen clear and CLI arguments example</p> <p><b>Modula-2</b></p> <p><b>trails</b> moving-worm graphics demo</p> <p><b>caseconvert</b> converts Modula-2 keywords to uppercase</p> <p><b>Forth</b> Bresenham circle algorithm example</p> <p><b>Analyze</b> 12 templates for the spreadsheet Analyze</p> <p>There are four programs here that read Commodore 64 picture files. They can translate Koala Pad, Doodle, Print Shop and News Room graphics to IFF format. Getting the files from your C-64 to your Amiga is the hard part.</p> <p><b>AMICUS Disk 12</b> <b>Executable programs</b></p> <p><b>blink</b> 'alink' compatible linker, but faster, E-D</p> <p><b>clean</b> spins the disk for disk cleaners, E-D</p> <p><b>epsonset</b> sends Epson settings to PAR from menu E-D</p> <p><b>showbig</b> view hi-res pics in low-res superbmap, E-D</p> <p><b>speaktme</b> tell the time, E-D</p> <p><b>undelete</b> undeletes a file, E-D</p> <p><b>crvaplphm</b> converts Apple II low, medium and high res pictures to IFF, E-D</p> <p><b>menued</b> menu editor produces C code for menus, E-D</p> <p><b>quick</b> quick-disk-to-disk nibble copier, E-D</p> <p><b>quickEA</b> copies Electronic Arts disks, removes protection, E-D</p> <p><b>demo</b> demo of text editor from Microsmiths, E-D</p> <p><b>rotating</b> rotating blocks graphics demo, S-E-D</p> <p><b>spn3</b> start a new CLI at the press of a button, like Sidekick, S-E-D</p> <p><b>popcli</b> VSprit example code from Commodore, S-E-D</p> <p><b>vsprite</b> VSprit example code from Commodore, S-E-D</p> <p><b>AmigaBBS</b> Amiga Basic bulletin board prog., S-D</p> <p><b>Assembler programs</b></p> <p><b>star10</b> makes star fields like Star Trek intro, S-E-D</p> <p><b>Pictures</b></p> <p><b>Mount Mandelbrot</b> 3D view of Mandelbrot set</p> <p><b>Star Destroyer</b> hi-res Star Wars starship</p> <p><b>Robot</b> robot arm grabbing a cylinder</p> <p><b>Texts</b></p> <p><b>vendors</b> Amiga vendors, names, addresses</p> <p><b>cardco</b> fixes to early Cardco memory boards</p> <p><b>cinclue</b> cross-reference to C include files</p> <p><b>mindwalker</b> clues to playing the game well</p> <p><b>slideshow</b> make your own slideshows from the Kaleidoscope disk</p> <p><b>AMICUS Disk 13</b> <b>AmigaBasic programs</b></p> <p>Routines from Carolyn Scheppner of CBM Tech Support, to read and display IFF pictures from Amiga Basic. With documentation. Also included is a program to do screen prints in Amiga Basic, and the newest BMAP files, with a corrected ConvertFD program. With example pictures, and the SaveILBM screen capture program.</p> <p>Routines to load and play FutureSound and IFF sound files from Amiga Basic, by John Foust for Applied Visions. With</p>	<p>documentation and C and assembler source for writing your own libraries, and interfacing C to assembler in libraries. With example sound.</p> <p><b>Executable programs</b></p> <p><b>gravity</b> Sci Amer Jan 86 gravitation graphic simulation, S-E-D</p> <p><b>Texts</b></p> <p><b>MIDI</b> make your own MIDI instrument interface, with documentation and a hi-res schematic picture.</p> <p><b>AMICUS Disk 14</b> <b>Several programs from Amazing Computing issues:</b></p> <p><b>Tools</b></p> <p><b>Dan Kary's</b> C structure index program, S-E-D</p> <p><b>Amiga Basic programs:</b></p> <p><b>BMAP Reader</b> by Tim Jones</p> <p><b>IFFBrush2BOB</b> by Mike Swinger</p> <p><b>AutoRequester</b> example</p> <p><b>DOSHelper</b> Windowed help system for CLI commands, S-E-D</p> <p><b>PETTrans</b> translates PET ASCII files to ASCII files, S-E-D</p> <p><b>C Squared</b> Graphics program from Scientific American, Sept 86, S-E-D</p> <p><b>crf</b> adds or removes carriage returns from files, S-E-D</p> <p><b>dpdecode</b> decrypts Deluxe Paint, remo</p> <p><b>ves copy</b> queryWB asks Yes or No from the user returns exit code, S-E</p> <p><b>vc</b> VisiCalc type spreadsheet, no mouse control, E-D</p> <p><b>view</b> views text files with window and slider</p> <p><b>Olmg, Spooling, YaBoing, Zoing</b> are sprite-based</p> <p><b>Boing!</b> style demos, S-E-D</p> <p><b>CLIClock, sClock, wClock</b> are window border clocks, S-E-D</p> <p><b>Texts</b></p> <p>An article on long-persistence phosphor monitors, tips on making brushes of odd shapes in Deluxe Paint, and recommendations on icon interfaces from Commodore-Amiga.</p> <p><b>AMICUS Disk 15</b> <b>The C programs include:</b></p> <p><b>'pr'</b> a file printing utility, which can print files in the background, and with line numbers and control character filtering.</p> <p><b>'tm'</b> displays a chart of the blocks allocated on a disk.</p> <p><b>'Aek'</b> questions an 'execute' file, returns an error code to control the execution in that batch file</p> <p><b>'Stat'</b> an enhanced version of AmigaDOS 'status' command.</p> <p><b>'Dissolve'</b> random-dot dissolve demo displays IFF picture slowly, dot by dot, in a random fashion.</p> <p><b>'PopCLI2'</b> invoke new CLI window at the press of a key.</p> <p><b>The executable programs include:</b></p> <p><b>'Form'</b> file formatting program through the printer driver to select print styles</p> <p><b>'DiskCat'</b> catalogs disks, maintains, sorts, merges lists of disk files</p> <p><b>'PSound'</b> SunRize Industries' sampled sound editor &amp; recorder</p> <p><b>'Iconmaker'</b> makes icons for most programs</p> <p><b>'Fractals'</b> draws great fractal seascapes and mountain scapes.</p> <p><b>'3D Breakout'</b> 3D glasses, create breakout in a new dimension</p> <p><b>'AmigaMonitor'</b> displays lists of open files, tasks, devices and ports in use.</p> <p><b>'Cosmoids'</b> version of 'asteroids' for the Amiga.</p> <p><b>'Sizzlers'</b> high resolution graphics demo written in Modula 2.</p> <p><b>Texts:</b></p> <p><b>'ansi.txt'</b> explains escape sequences the CON: device responds to.</p> <p><b>'FKey'</b> includes template for making paper to sit in the tray at the top of the Amiga keyboard.</p> <p><b>'Spawn'</b> programmer's document from Commodore</p> <p><b>Amiga</b> describes ways to use the Amiga's multitasking capabilities in your own programs.</p> <p><b>AmigaBasic programs:</b></p> <p><b>'Grids'</b> draw sound waveforms, and hear them played.</p> <p><b>'Light'</b> a version of the Tron light-cycle video game.</p> <p><b>'MigaSol'</b> a game of solitaire.</p> <p><b>'Stats'</b> program to calculate batting averages</p> <p><b>'Money'</b> "try to grab all the bags of money that you can."</p> <p><b>AMICUS 15</b> also includes two beautiful IFF pictures, of the enemy walkers from the ice planet in Star Wars, and a picture of a cheetah.</p> <p><b>AMICUS Disk 16</b> <b>'Juggler'</b> demo by Eric Graham, a robot juggler bouncing three mirrored balls, with sound effects. Twenty-four frames of HAM animation are flipped quickly to produce this image. You control the speed of the juggling. The author's documentation hints that this program might someday be available as a product.</p> <p><b>IFF pictures</b></p> <p>parodies of the covers of Amiga World and Amazing Computing magazines.</p> <p><b>C programs:</b></p> <p><b>'inputhandler'</b> example of making an input handler.</p> <p><b>'FileZap3'</b> binary file editing program</p> <p><b>'ShowPrint'</b> displays IFF pictures, and prints it.</p> <p><b>'Gen'</b> program indexes and retrieves C structures and variables declared in the Amiga include file system.</p> <p><b>Executable Programs:</b></p> <p><b>'FixHunk2'</b> repairs an executable program file for expanded memory</p> <p><b>'ms2mus'</b> converts Music Studio files to IFF standard 'SMUS' format. I have heard this program might have a few bugs, especially in regards to very long songs, but it works in most cases.</p> <p>Amiga version of the Missile Command video game,</p>	<p>This disk also contains several files of scenarios for Amiga Flight Simulator II. By putting one of these seven files on a blank disk, and inserting it in the drive after performing a special command in this game, a number of interesting locations are preset into the Flight Simulator program. For example, one scenario places your plane on Alcatraz, while another puts you in Central Park</p> <p><b>AMICUS Disk 17</b> <b>Telcommunications disk</b> which contains six terminal programs.</p> <p><b>'Comm' V1.33</b> term prog. with Xmodem, WModem, term prog. includes Super Kermit</p> <p><b>'ATerm' V7.2</b> Dave Wecker's VT-100 emulator with Xmodem, Kermit, and scripting</p> <p><b>'VT-100' V2.6</b> V4D(060) port of the Unix C-Kermit</p> <p><b>'Amiga Kermit'</b> Tektronix graphics terminal emulator based on the VT-100 prog. V2.3 and contains latest 'arc' file compression</p> <p><b>'AmigaHost'</b> V0.9 for CompuServe. Includes RLE graphics abilities &amp; CIS-B file transfer protocol.</p> <p><b>'FixHunk'</b> expansion memory necessity</p> <p><b>'FixObj'</b> removes garbage characters from modern received files</p> <p><b>'Txt'</b> filters text files from other systems to be read by the Amiga E.C.</p> <p><b>'addmem'</b> executable version for use with mem expansion article in AC v2.1</p> <p><b>'arc'</b> file documentation and a basic tutorial on an 'arcing' files for making 'arc' files E.C.</p> <p><b>AMICUS Disk 18</b> <b>Logo</b></p> <p>Amiga version of the popular computer language, with example programs, E-D</p> <p><b>TVText</b> Demo version of the TVText character generator</p> <p><b>PageSetter</b> Freely distributable versions of the updated PagePrint and PageIFF programs for the PageSetter desktop publishing package.</p> <p><b>FullWindow</b> Resizes any CLI window using only CLI commands, E-D</p> <p><b>Lite3d</b> 3-D version of Conway's LIFE program, E-D</p> <p><b>Defdisk</b> CLI utility to re-assign a new Workbench disk, S-E-D</p> <p><b>Calendar.WKS</b> Lotus-compatible worksheet that makes calendars</p> <p><b>SetKey</b> Demo of keyboard key re-programmer, with IFF picture to make function key labels, E-D</p> <p><b>VPG</b> Video pattern generator for aligning monitors, E-D</p> <p><b>HP-10C</b> Hewlett-Packard-like calculator, E-D</p> <p><b>SetPrefs</b> Change the Preferences settings on the fly, in C, S-E-D</p> <p><b>StarProbe</b> Program studies stellar evolution.</p> <p><b>C source</b> included for Amiga and MS-DOS, S-E-D</p> <p><b>ROT</b> C version of Colin French's AmigaBasic ROT program from Amazing Computing. ROT edits and displays polygons to create three dimensional objects. Up to 24 frames of animation can be created and displayed. E-D</p> <p><b>Scat</b> Like ling, windows on screen run away from the mouse, E-D</p> <p><b>DK</b> Decays" the CLI window into dust, in Modula 2, S-E-D</p> <p><b>DropShadow2</b> Adds layered shadows to Workbench windows, E-D</p> <p><b>AMICUS Disk 19</b> <b>This disk carries several programs from Amazing Computing.</b> The IFF pictures on this disk include the Amiga Wake part T-shirt logo, a sixteen-color hi-res image of Andy Griffith, and five Amiga Live! pictures from the Amazing Stories episode that featured the Amiga.</p> <p><b>Solve</b> Linear equation solver in assembly language, S-E-D</p> <p><b>Gadgets</b> Bryan Catley's AmigaBasic tutorial, Bryan Catley's AmigaBasic household inventory program, S-D</p> <p><b>Household</b> John Shields' Waveform WorkbenchBasic, S-D</p> <p><b>Waveform</b> Jim Kennan's AmigaBasic disk librarian program, S-D</p> <p><b>DKLib</b> Ivan Smith's AmigaBasic subscript example, S-D</p> <p><b>Subscripts</b> C programs and executables for Harriet Maybeck Tolly's Intuition tutorials, S-E-D</p> <p><b>String, Boolean</b></p> <p><b>Skinny C</b> Bob Riemersma's example for making small C programs, S-E-D</p> <p><b>COMAL.h</b> Make C look like COMAL header file, Makes Emacs function key definitions by Greg Douglas, S-D</p> <p><b>EmacsKey</b> Snoop on system resource use, E-D</p> <p><b>Amon 1.1</b> Bard's Tale character editor, E-D</p> <p><b>BTE</b> CLI program shows the size of a given set of files, E-D</p> <p><b>Size</b> CLI window utility resizes current window, S-E-D</p> <p><b>WinSize</b></p> <p><b>AMICUS Disk 20</b> <b>Compactor, Decoder</b> Steve Michel AmigaBasic tools, S-D</p> <p><b>BobEd</b> BOB and sprite editor written in C-S-E-D</p> <p><b>SpriteMasterII</b> Sprite editor and animator by Brad Kiefer, E-D</p> <p><b>BlitLab</b> Blitter chip exploration C program by Tomas Rokicki, S-E-D</p> <p><b>FFPIC</b> Image processing program by Bob Bush loads and saves IFF images, changes them with several techniques, E-D</p> <p><b>Bankn</b> Complete home banking program, balance your checkbook! E-D</p> <p><b>AMICUS Disk 21</b> <b>Target</b></p> <p>Makes each mouse click sound like a gunshot, S-E-D</p> <p><b>Sand</b> Simple game of sand that follows the mouse pointer, E-D</p>	<p><b>PropGadget</b> Harriet Maybeck Tolly's proportional gadget example, S-E</p> <p><b>EBB</b> Checks to see if you have extra-half-bright graphics, S-E-D</p> <p><b>Piano</b> Simple piano sound program</p> <p><b>CelScripts</b> Makes cel animation scripts for Aegis Animator, in AmigaBasic</p> <p>This disk has electronic catalogs for AMICUS disks 1 to 20 and Fish disks 1 to 80. They are viewed with the DiskCat program, included here.</p> <p><b>AMICUS Disk 22</b> <b>Cycles</b></p> <p><b>Show_Print</b> Light cycle game, E-D</p> <p><b>Views and prints</b> IFF pictures, including larger than screen</p> <p><b>PrtDrvGen2.3</b> Latest version of a printer driver generator</p> <p><b>Animations</b> VideoScope animations of planes and boing ball</p> <p><b>Garden</b> Makes fractal gardenscapes</p> <p><b>BasicSorts</b> Examples of binary search and insertion sort in AmigaBasic</p> <p><b>AMICUS Disk 23</b> <b>An AMICUS disk completely dedicated to music on the Amiga.</b> This disk contains two music players, songs, instruments, and players to bring the thrill of playing "Big Sound" on your Amiga.</p> <p><b>Instruments</b> a collection of 25 instruments for playing and creating music. The collection ranges from Cannon to Marimba</p> <p><b>List INSTR</b> program to list the instruments DMCS will not load as well as list the origins for any instrument.</p> <p><b>Music</b> a collection of 14 Classical pieces</p> <p><b>1612Overture</b> The 16 minute classical piece complete with Cannon!</p> <p><b>Three Amiga Music Players:</b></p> <p><b>SMUSPlay</b></p> <p><b>MusicCraft2SMUS</b></p> <p><b>MusicStudio2SMUS</b></p> <p><b>AMICUS Disk 24</b> <b>Sectorama</b></p> <p>A disk sector editor for any AmigaDOS file-structured device, recover files from a trashed hard disk. By David Joiner of MicroIlusions</p> <p><b>Iconize</b> Reduces the size of IFF images, companion program. Recolor, remaps the palette colors of one picture to use the palette colors of another. Using these programs and a tool to convert IFF brushes to Workbench icons, make icons look like miniatures of the pictures.</p> <p><b>CodeDemo</b> Module-2 program converts assembler object files to inline CODE statements. Comes with a screen scrolling example</p> <p><b>AmiBug</b> Workbench hack makes the same fly walk across the screen at random intervals. Otherwise, completely harmless.</p> <p><b>BNTools</b> Three examples of assembly language code from Bryce Nesbitt:</p> <ol style="list-style-type: none"> <li>1. SetLace.prog to switch interface on/off.</li> <li>2. Why, replace AmigaDOS CLI Why</li> <li>3. Loadit, prog to load a file into memory until a reboot. (Only the most esoteric hackers will find Loadit useful.)</li> </ol> <p><b>Monolace</b> CLI program resets Preferences to several colors of monochrome &amp; interface screens. C source is included, works with DisplayPref, a CLI program which displays the current Preferences settings.</p> <p><b>BoingMachine</b> A ray-traced animation of a perpetual motion Boing-making machine, includes the latest version of the Movie program, which has the ability to play sounds along with the animation. By Ken Offer</p> <p><b>Daisy</b> Example of using the translator and narrator devices to make the Amiga talk. It is written in C.</p> <p><b>QuickFlx</b> Script-driven animation and slideshow program flips through IFF images.</p> <p><b>BMon</b> System monitor AmigaBasic program; perform simple manipulations of memory.</p> <p><b>Moose</b> Random background program, a small window opens with a moose resembling Bullwinkle saying witty phrases user definable.</p> <p><b>DGCS</b> Deluxe Grocery Construction Set, simple Intuition-based prog for assembling and printing a grocery list.</p> <p><b>The Virus Check</b> directory holds several programs relating to the software virus that came to the US from pirates in Europe as detailed in Amazing Computing V2.12. Bill Koester's full explanation of the virus code is included. One program checks for the software virus on a Workbench disk; the second program checks for the virus in memory, which could infect other disks.</p> <p><b>AMICUS Disk 25</b> <b>Nemesis</b></p> <p>Graphics demo pans through space towards the mythical dark twin of the sun with wonderful music and space graphics.</p> <p><b>The KickPlay</b> directory holds text that describes several patches to the Kickstart disk. For Amiga 1000 hackers who feel comfortable patching a disk in hexadecimal, KickPlay offers the chance to automatically do an ADDMEM for old expansion memory, as well as the ability to change the picture of the "Insert Workbench" hand. A program is also included for restoring the correct checksum of the Kickstart disk.</p> <p><b>KeyBird</b> BASIC prog edits keymaps, adjust the Workbench keymaps or create your own.</p>
---	--	---	--



8ColorWB	Modifies the Workbench so three bitplanes are used, icons can have eight colors, instead of four, eight-color icons are included. Public domain program "zapicon" or "brush2icon" converts eight-color IFF brushes to icons, to use Deluxe Paint to make icons for this new Workbench.	<b>Fred Fish Disk 2:</b> aib cc dbug make make2 microemac portar xrl <b>Fred Fish Disk 3:</b> gothic roff ft clorth xlisp <b>Fred Fish Disk 4:</b> banner bgrep blson bm grip kermil MyCLI mandel <b>Fred Fish Disk 5:</b> cons freemap input.dev joystick keyboard layers mandelbrot mouse one.window parallel printer print.support proctest region samplefont serial singlePlayfield speechdemo textdemo timer trackdisk <b>Fred Fish Disk 6:</b> compress dadc microemac mult scales setparallel setserial setstrip stpic <b>Fred Fish Disk 7:</b> This disk contains the executables of the game Hack V 1.0.1. <b>Fred Fish Disk 8:</b> This disk contains the C source to Hack on disk 7. <b>Fred Fish Disk 9:</b> moire MVP-FORTH a more powerful text formatting program Prog to toggle interface mode on and off a rubic's cube type demo moving snake Graphics demo <b>Fred Fish Disk 10:</b> conquest dehex filezap fixobj ift ld ls squsq trek73 yacht <b>Fred Fish Disk 11:</b> dpslide <b>Fred Fish Disk 12:</b> amiga3d sign ArgoTerm arrow3d ld4 iconExec SetWindow SetAlternate StarTerm	Object module librarian. Unix-like frontend for Lattice C compiler. Macro based C debugging package. Machine independent. Subset of Unix make command. Another make subset command. Small version of emacs editor, with macros, no extensions Portable file archiver. DECUS C cross reference utility. Gothic font banner printer. A "roff" type text formatter. A very fast text formatter A highly portable forth implementation. Lots of goodies. Xlisp 1.4, not working correctly. Prints horizontal banner A Boyer-Moore grep-like utility GNU Unix replacement "yacc", not working. Another Boyer-Moore grep-like utility DECUS grep simple portable Kermit with no connect mode. Replacement CLI for the Amiga. V. 1.0 A Mandelbrot set program, by Robert French and RJ Mical Console device demo program with supporting macro routines. Creates a visual diagram of free memory sample input handler, traps key or mouse events Shows how to set up the gameport device as a joystick. demonstrates direct communications with the keyboard. Shows use of the layers library IFF Mandelbrot program hooks up mouse to right joystick port console window demo Demonstrates access to the parallel port. opening and using the printer, does a screen dump, not working. Printer support routines, not working. sample process creation code, not working demos split drawing regions sample font with info on creating your own Demos the serial port Creates 320 x 200 playfield latest version of cute speech demo simplified version of speechdemo, with IO requests displays available fonts demos timer.device use demos trackdisk driver like Unix compress, a file squeezer analog clock impersonator upgraded version of microemac from disk 2 removes multiple occurring lines in files demos using sound and audio functions Allows changing parallel port parameters Allows changing serial port parameters. quicksort based sort program, in C Strips comments and extra whitespace from C source An interstellar adventure simulation game convert a hex file to binary Patch program for any type of file. Strip garbage off Xmodem transferred files. Routines to read and write IFF format files. simple directory program Minimal UNIX ls, with Unix-style wildcarding, in C file squeeze and unsqueeze Star Trek game Dico game. slide show program for displaying IFF images with miscellaneous pictures Shows a rotating 3 dimensional solid "Amiga sign". a terminal emulator program, written in assembler Shows a rotating 3 dimensional wire frame arrow. directory listing program two progs for launching progs from Workbench, presently only works under CLI. Makes an icon show a second image when clicked once terminal emulator, with ASCII Xmodem, dialer, more.	<b>Fred Fish Disk 13:</b> A Bundle of Basic programs, including: Jpad toybox xmodem ror bounce cardit cubes1 dragon Eliza fscap hail9000 join minipaint pena Readme sabotage shuttle sketchpad speechy spiral talk termtest wheels xenos (note: some programs are Abasic, most are AmigaBASIC, and some programs are presented in both languages) <b>Fred Fish Disk 14:</b> amiga3d beep dex dimensions filezap gtxmem gi pcterm shell termcap <b>Fred Fish Disk 15:</b> Blobs Clock Dazzle Fish Monopoly OkidataDump Polydraw Polyfractals <b>Fred Fish Disk 16:</b> A complete copy of the latest developer IFF disk <b>Fred Fish Disk 17:</b> The NewTek Digi-View video digitizer HAM demo disk <b>Fred Fish Disk 18:</b> AmigaDisplay Ash Browser MC68010 Multidim PigLatin Scrimper Xlisp1.6 <b>Fred Fish Disk 19:</b> BlackJack JayMinerSlides Keymap_Test LockMon <b>Fred Fish Disk 20:</b> AmigaToAtari DiskSalv Hash Hd MandelBrot MultiTasking Pack PortHandler Random SetMouse2 SpeechTerm TxEd <b>Fred Fish Disk 21:</b> This is a copy of Thomas Wilcox's Mandelbrot Set Explorer disk. Very good! <b>Fred Fish Disk 22:</b> This disk contains two new "strains" of microemac. version 3.5 by Daniel Lawrence. For Unix V7, BSD 4.2, Amiga, MS-DOS, VMS. Uses Amiga function keys, status line, execute, startup files, more. By Andy Poggio. New features include <ALT> keys as Meta keys, mouse support, higher priority, backup files, word wrap, function keys. ezspeak adbook adbook-copy amiga-copy brkout colorcircles datedogstar dynamictangle fillbuster dart hauntedM hail9000 loz mouse pinwheel rgb shades speakspeech sphere superpad supstr triangle topography xmstrip Amiga Basic demos; Carolyn Schepner. finds .bmap from IFF files. creates addresses of and writes to bitplanes of the screen's bmap. A tutorial on creation and use of bmaps. loads and displays IFF ILM pics. loads and displays ACBM pics. creates a demo screen and dumps it to a graphic printer. Simple 68000 disassembler. Reads standard Amiga object files and disassembles the code sections. Data sections are dumped in hex. The actual disassembler routines are set up to be callable from a user prog so instructions in memory can be disassembled dynamically. By Bill Rogers. Example of a keypad structure for the Dvorak keyboard layout. Untested but included because assembly examples are few and far between. By Robert Burris Spiragraph, from Feb. 84 Byte. Example of proportional gadgets to scroll a SuperBMap. Schematics and directions for building your own homebrew 1 Mo memory expansion, by Michael Feilinger. Program to debug "halloic" calls Convert Julian to solar and sidereal time, stellar positions and radial velocity epoch calculations and Galilean satellite plotter. By David Eagle. Abasic games by David Addison: Backgammon, Cribbage, Milestone, and Othello DECUS 'tpp' C preprocessor, & a modified 'cc' that knows about the 'tpp', for Marc C. Unix-compatible shell archiver, for packing files for travel. Example of using a ScrollLayer, syncing SuperBMap for printing, and creating dummy RastPorts. AegisDraw Demo Demo program without save and no docs. Animator Demo Player for the Aegis Animator files Cc Unix-like front-end for Mana C Tests for existence of system resources, files, and devices Animated Rubik's cube program VT-100 terminal emulator with Kermit and Xmodem protocols Several shareware programs. The authors request a donation if you find their program useful, so they can write more software. an Amiga Basic BBS by Ewan Grantham Amiga art edit fonts, by Tim Robinson Create menus, save them as C source, by David Peterson Very nice telecom. by J. Nangano (Fred Fish Disk#30 is free if requested when ordered with at least three other disks from the collection.) Life game, uses blitter to do 19.8 generations a second. Version 3.0 of Robert French's program. Mutual exclusion gadget example. Measure relative RAM speed, chip and fast. Replacement for the Marx "set" command for environment variables, with improvements. Draws a recursive tree, green leafy type, not files. Crippled demo version of Microsmith's text editor, TxEd. Full-featured drawing program by Stephen Vermeulen. Invokes CLI scripts from icon Displays text files from an icon.	<b>Fred Fish Disk 23:</b> Disk of source for MicroEmacs, several versions for most popular operating systems on micros and mainframes. For people who want to port MicroEmacs to their favorite machine. <b>Fred Fish Disk 24:</b> Conquest Csh Module-2 Module-2 compiler originally developed for Macintosh at ETHZ. This code was transmitted to the AMIGA and is executed on the AMIGA with a special loader. Binary only. <b>Fred Fish Disk 25:</b> Graphic Hack A graphic version of the game on disks 7 and 8. This is the graphics-oriented Hack game by John Toebes. Only the executable is present. <b>Fred Fish Disk 26:</b> UnLink Collect code, data, and bss hunks together, allows individual specification of code, data, and bss origins, and generates binary file with format reminiscent of Unix "a.out" format. The output file can be easily processed by a separate program to produce Motorola "S-records" suitable for downloading to PROM programmer. By Eric Black. C-kermit Port of the Kermit file transfer program and server. Display and set process priorities Yet another program for bundling up text files and mailing or posting them as a single file unit. <b>Fred Fish Disk 27:</b> ABSDemos NewConventFD BitPlanes AboutBmaps LoadIBM LoadACBM ScreenPrint Disassem DvorakKeymap Hypocycloids LinesDemo MemExpansion SaleMailoc ScienceDemos <b>Fred Fish Disk 28:</b> Abasic games by David Addison: Backgammon, Cribbage, Milestone, and Othello C++ DECUS 'tpp' C preprocessor, & a modified 'cc' that knows about the 'tpp', for Marc C. Unix-compatible shell archiver, for packing files for travel. Example of using a ScrollLayer, syncing SuperBMap for printing, and creating dummy RastPorts. <b>Fred Fish Disk 29:</b> AegisDraw Demo Demo program without save and no docs. Animator Demo Player for the Aegis Animator files Cc Unix-like front-end for Mana C Tests for existence of system resources, files, and devices Animated Rubik's cube program VT-100 terminal emulator with Kermit and Xmodem protocols Several shareware programs. The authors request a donation if you find their program useful, so they can write more software. an Amiga Basic BBS by Ewan Grantham Amiga art edit fonts, by Tim Robinson Create menus, save them as C source, by David Peterson Very nice telecom. by J. Nangano (Fred Fish Disk#30 is free if requested when ordered with at least three other disks from the collection.) <b>Fred Fish Disk 31:</b> Life game, uses blitter to do 19.8 generations a second. Version 3.0 of Robert French's program. Mutual exclusion gadget example. Measure relative RAM speed, chip and fast. Replacement for the Marx "set" command for environment variables, with improvements. Draws a recursive tree, green leafy type, not files. Crippled demo version of Microsmith's text editor, TxEd. Full-featured drawing program by Stephen Vermeulen. Invokes CLI scripts from icon Displays text files from an icon.
----------	--	---	---	---	--

### Fred Fish Public Domain Software

#### Fred Fish Disk 1:

amigademo  
amigaterm  
balls  
colorful  
dhrystone  
doty  
freedraw  
gad  
gtxmem  
halibrite  
hello  
latip  
palette  
trackdisk  
requesters  
speech  
speechtoy

Graphical benchmark for comparing amigas.  
simple communications program with Xmodem  
simulation of the "kinetic things" with balls on strings  
Shows off use of hold-and-modify mode.  
Dhrystone benchmark program.  
Source to the "doty window" demo on the Workbench disk.  
A small "paint" type program with lines, boxes, etc.  
John Draper's Gadget tutorial program  
Graphical memory usage display prog.  
demonstrates "Extra-Half-Brite" mode, if you have it  
simple window demo  
accessing the Motorola Fast Floating Point library from C  
Sample prog. to design color palettes.  
Demonstrates use of the trackdisk driver.  
John Draper's requester tutorial and example program.  
Sample speech demo program.  
Stripped down "speechtoy".  
Another speech demo program.



<p><b>Fred Fish Disk 32</b>          Address Extended address book, AmigaBASIC          Calendar Calendar/diary program, AmigaBASIC          DosPlus1 First volume of CLI oriented developer tools          DosPlus2 2nd volume of CLI oriented developer tools.          Executables only:          MacView Views MacPaint pics in Amiga low or high res, no sample pictures, by Scott Evernden.          Puzzle Simulation of puzzle with moving squares.          ShowHAM View HAM pictures from CLI.          Solitaire ABASIC games of Canfield and Klondike, from David Addison.          Spir3 Graphics demo of spinning cubes, double-buffered example.          Sword Sword of Fallen Angel text adventure game written in Amiga Basic.          Trails Leaves a trail behind mouse, in Modula-2</p> <p><b>Fred Fish Disk 33</b>          3dstars 3d version of the "stars" program below.          Bignap Low-level graphics example scrolls bitmap with ScrollVPort.          Dbl.gels Double-buffered animation example for BOBs and VSprites.          DiskMapper Displays sector allocation of floppy disks.          MenView View memory in real time, move with joystick.          Spring Bouncing balls demo.          ScreenDump Oups, with sound effects.          Sub Dumps highest screen or window to the printer.          Stars Simple database program from a DECUS tape.          Stars Star field demo, like Star Trek.          TermPlus Terminal program with capture, library, function keys, Xmodem, C-IS-B protocols.          Vt100 Version 2.0 of Dave Wecker's VT-100 emulator, with scripts &amp; function</p> <p><b>Fred Fish Disk 34</b>          Alint Support files for Gimpel's 'lint' syntax checker          Blink PD 'alink' compatible linker, faster, better.          Browser Updated to FF 18 'browser', in Marx, with scroll bars, bug fixes.          Btree b-tree data structure examples          Btree2 Another version of 'btree'          Calendar Appointment calendar with alarm.          Less File viewer, searching, position by percent, line number.          NewFonts Set of 26 new Amiga fonts from Bill Fischer          Pr Background print utility, style options, wildcards.          Requester Deluxe Paint-type file requester, with sample.</p> <p><b>Fred Fish Disk 35</b>          ASendPacket C example of making asynchronous I/O calls to a DOS handler, written by C-A          ConsoleWindow C example of getting the intuition pointer to a CON or RAW window, for 1.2, by C-A.          DirUtil Walk the directory tree, do CLI operations from menus          DirUtil2 Another variant of DirUtil.          FileRequester Lattice C file requester module, with demo driver, from Charlie Heath.          MacView Views MacPaint pictures in Amiga low or high res, with sample pictures, by Scott Evernden.          Pop Simple IFF reader program          PopCLI Sidelink-style program invokes a new CLI, with automatic screen blanking.          QuickCopy Devlopent disk copiers duplicate copy-protected disks.          ScrollP Dual playfield example, from C-A shows 400 x 300 x 2 bit plane playfield on a 320 x 200 x 2 plane deep playfield.          SendPacket General purpose subroutine to send AmigaDOS packets.          SpriteMaker Sprite editor, can save work as C data structure. Shareware by Ray Larson.          Tracker Converts any disk into files, for electronic transmission. Preserves entire file structure. Shareware by Brad Wilson.          TriClops 3-D Space invasion game, formerly commercial, now public domain. From Geodesic Publications.          Tsize Print total size of all files in subdirectories.          Unifdel C preprocessor to remove given #ifdef sections of a file, leaving the rest alone. By Dave Yost          Vtest VT-100 emulation test program. Requires a Unix system.</p> <p><b>Fred Fish Disk 36</b>          Acp Unix-like 'cp' copy program          Clock Updated version of clock on disk 15.          Csh Marx 'csh'-like CLI, history, variables, etc.          DietAid Diet planning aid organizes recipes, calories          Echo Improved 'echo' command with color, cursor addressing          FixHunk Fixes programs to let them run in external memory.          Fm Maps the sectors a file uses on the disk.          KokBench Docs, program to make a single disk that works like a Kickstart and Workbench.          Lex Computes Fog, Fleisch, and Kincaid readability of text files.          TunnelVision David Addison ABASIC 3D maze perspective game.          Vc Visicalc-like spreadsheet calculator program.          Vt100 Version 2.2 of Dave Wecker's telecom program          YaBoing Oingo! style game program shows sprite collision detection</p> <p><b>Fred Fish Disk 37</b>          This disk is a port of Timothy Budd's Little Smalltalk system, done by Bill Kinnerley at Washington State University.</p> <p><b>Fred Fish Disk 38</b>          CSquared Sep 86 Sci American, Circle Squared algorithm          FixQty Strips garbage off Xmodem transferred object files          Handler AmigaDOS handler (device) example from C-A</p>	<p>Hp-10c Mimics a HP-10C calculator, written in Modula-2          IFFEncode Saves the screen as an IFF file          JiffDump Dumps info about an IFF file          JiffMulti BDS C-like CLI shell          JiffStatus STATUS-like program, shows priority, processes          JiffReversi Game of Reversi, version 6.1          Ucodecode Translate binary files to text, Unix-like programs          Vdraw Drawing program, version 1.14          VoiceFilter DX MIDI synthesizer voice filter program          Window Example of creating a DOS window on a custom screen</p> <p><b>Fred Fish Disk 39</b>          AnsiEcho 'echo', 'touch', 'list', 'dis' written in assembler.          Display Displays HAM images from a ray-tracing program, with example pictures.          Driver Example device driver source, acts like RAM: disk          Xlisp 1.7, executable only</p> <p><b>Fred Fish Disk 40</b>          Ahost Terminal emulator with Xmodem, Kermit and C-IS-B protocols, function keys, scripts, RLE graphics and conference mode.          AmigaMonitor Dynamically displays the machine state, such as open files, active tasks, resources, device states, interrupts, libraries, ports, etc.          Arc Popular file compression system, the standard for transmitting files          AreaCode Program that decodes area codes into state and locality.          Blink 'alink' replacement linker, version 6.5          Cosmo An 'asteroids' clone.          Dg210 Data General D-210 Terminal emulator          DirUtil Windowed DOS interface program, V.1.4          DOSHelper Windowed AmigaDOS CLI help program          PagePrint Prints text files with headers, page breaks, line numbers</p> <p><b>Fred Fish Disk 41</b>          PopCLI Starts a new CLI with a single keystroke, from any program. With a screen-saver feature. Version 2, w/ resource.          SpriteEdit Sprite Editor edits two sprites at a time          X-Spell Spelling checker allows edits to files</p> <p><b>Fred Fish Disk 42</b>          AmigaVenture Create your own text adventure programs in AmigaBASIC.          Version 2.03 of Dillon's C-sh-like shell.          Csh Executable only          Dbug Macro based C debugging package, updated to FF 2          DualPlayfield example from CBM, update to intuition manual          GetFile Heath's file requester, with source          LatXref Cross reference of Lattice 3.10 header files          Lines Line drawing demo program          SelfFont Changes font used in a CLI window          Vt100 Version 2.3 of the VT-100 terminal program.</p> <p><b>Fred Fish Disk 43</b>          BasicBoling This disk contains an Amiga version of MicroGnuEmacs.          BasicBoling AmigaBasic program demos page flipping of a 3D cube          Bbm Demo copy of B.E.S.T. Business Management System.          BbmList A list of Amiga Bulletin Board Systems          Cc C compiler frontends for Marx and Lattice C          Copper A hardware copper list disassembler          InstIFF Converts Instruments demo sounds to IFF sampled sounds          PopColors Adjust RGB colors of any screen          SpriteClock Simple clock is displayed on a sprite above all screens          ST Emulator Non-serious Atari ST emulator          WBrn Lets Workbench programs be run from the CLI          Wild Two Unix shell style wild card matching routines</p> <p><b>Fred Fish Disk 44</b>          Icons Miscellaneous icons          NewIFF New IFF material from CBM for sampled voice and music files          RayTracePics The famous ray-tracing pictures, from FF39, now converted to IFF HAM format for 'much' faster viewing.          ViewILBM Displays normal and HAM ILBM files</p> <p><b>Fred Fish Disk 45</b>          Cue Cue board game          Make Another 'make', with more features          Pictures Miscellaneous pictures          Updates Updates older disk with newer files from another disk          WhereIs Searches a disk for files of given name</p> <p><b>Fred Fish Disk 46</b>          Asm Shareware 68010 macro assembler, ROM          Kernel Manual compatible          'execute' file program detects presence of modem          Gadget editor from the Programmers Network          Transforms a file from English to Live.          MyLib A binary only copy of Ma's alternate runtime library. Author: Matt Dillon          Subset Berkeley 'ms' and 'nm' macros for 'prof'          Transforms a file from English to Valley Speak.</p> <p><b>Fred Fish Disk 47</b>          3D-Arm Simulation of a robotic arm, very good graphics, teaching tool, including C source.          Juggler Eric Graham's stunning HAM animation of a robot juggler          VT-100 Version 2.4 of Dave Wecker's terminal emulator, with Xmodem and Kermit file transfer protocols</p> <p><b>Fred Fish Disk 48</b>          Bru Alpha version of a hard disk file archiver          Version 1.30 of a terminal emulator with phone directories          Version 2.04 of Matt Dillon's Unix 'csh'-like CLI replacement, including Lattice &amp; Marx C source          Disk benchmark program for Unix and Amiga          Computes disk storage of a file or directory          Program to watch for programs that trash low memory. It attempts to repair the damage, and puts up a requester to inform you of the damage. From the Software Distillery.          A realtime execution profiler for Marx C programs. Includes C source.</p>	<p><b>Fred Fish Disk 49</b>          Cycloids Update of electronic spirograph from disk 27          DirUtil Enhanced version of DirUtil from disk 35          MultiDef Scans a set of object modules and libraries searching for multiply defined symbols          MyUpdate Disk update utility with options for stripping comments from C header files, and interactive verification of the updating process          Computes and displays 3 dimensional functions in hires          Polygon Moire type pattern generator with color cycling          QMouse Queries whether a mouse button is pressed. This can give a return code that can customize a startup-sequence based on whether a mouse button was pressed.          Touch Example of setting the datestamp on a file, using a technique from Commodore-Amiga          More extensive version of the trees program on disk 31</p> <p><b>Fred Fish Disk 50</b>          Asm Version 1.1 of a shareware 68000 macro assembler, compatible with the Metacomco assembler. This includes an example startup module and more Motorola mnemonics.          BreakOut A brick breakout game, uses 3-D glasses          DiskZip Version 1.1 of a program to edit disks and binary files          FirstSilicon A smart CLI replacement with full editing and recall of previous commands          A Missile Command-type game, with sound, in assembler          PerfectSound Sound editor for a low-cost sound digitizer          Sizzlers Graphics demos          UnixArc Ver. of 'arc' for Unix System V machines, in C          Wombat Version 3.01 of Dave Warkner's terminal emulator</p> <p><b>Fred Fish Disk 51</b>          Bison GNU for Unix 'yacc', working update to FF4          Compress Update to the file compression program on disk 6          Cos 'Wheel of Fortune'-type game in AmigaBASIC          DfSeed Unix 'df' and 'ssd' for finding the differences between two files, and then recreating the other, given one file, and the list of differences.          Portable versions of the CPM squeeze and unsqueeze</p> <p><b>Fred Fish Disk 52</b>          Assign Replacement for AmigaDOS 'assign' command in C          Fractal Makes random fractal terrains          Poly, HAMPoly Workbench demo programs for making polygons in hires and HAM          Example of mutual exclusion gadgets with GadgetText          Tek4010 Tektronix 4010 terminal emulator          VDraw Versions 1.16 and 1.19 of a Deluxe Paint-like drawing program</p> <p><b>Fred Fish Disk 53</b>          Animations Demo animations with player program for Aegis Animator          ARCre Creates rename scripts for files with long names, so they can be easily 'arced' and 'un-arc'd'.          Preliminary AmigaDOS replacements for 'break', 'cd', 'chmod', 'echo', 'filetype' and 'mkdir'          Compiler Not fully ported to the Amiga, this is a 68000 C compiler. It will produce simple assembly language output, but needs a lot of work.          Spreadsheet Updated with source of the 'vc' spreadsheet on disk 36          TarSplit Port of program to split Unix tar archives          Uuencode Utilities to encode and decode binary files for ASCII transmission, expanding them by 35%</p> <p><b>Fred Fish Disk 54</b>          Hanoi Solves Towers of Hanoi problem in it's own Workbench window. by Al Ozer          ISpell Port of a Unix screen oriented, interactive spelling checker. (Expansion RAM required) by Pace Willison          Ing A Screen of lots of bouncing tile windows by 'Bols Ewhac' Schwab          Displays number of tasks in run queue, averaged over last 1, 5, and 15 minute periods, by William Rucklidge          MIDITools Programs to play/record through the MIDI IFF, by Fred Cassiner          MoreRows Program to make the Work Bench Screen larger than normal, by Neil Katin and Jim Mackraz          Tilt Program to make your Amiga look like it didn't pass vibration testing, by Leo 'Bols Ewhac' Schwab</p> <p><b>Fred Fish Disk 55</b>          Csh V2.05 of Matt Dillon's csh like shell (Modified for Marx C), by Matt Dillon.          Modified by Steve Drew          NewStartups New C Startup modules:          ASStartup with 1.2 fires and better quote handling, opens a side window, using user specs, by Commodore,          posted to BIX by Carolyn Scheppe          Change another program's screen colors, by Carolyn Scheppe          Palette Allows the standard output of one process to be fed to the standard input of another, by Matt Dillon          Save a normal or HAM mode screen as an IFF file, by Carolyn Scheppe          Demo of the Activation game Shanghai.          A double buffered text example for Marx C, by Jim Goodnow          A working vsprite example, by Eric Cotton</p>	<p>Vt100 V2.6 of Dave's Vt100 terminal emulator with kermit and xmodem, by Dave Wecker</p> <p><b>Fred Fish Disk 56</b>          ClipBoard Clipboard device interface routines, to provide a standard interface. By Andy Finkle          Demos the use of DOS Packets, ConUnit, etc. by Carolyn Scheppe          Program to find all available disk device names and return them as an exec list, by Philip Lindsay          GetVolume Program to get volume name of the volume that a given file resides on, by Chuck McManis          Icon2C Reads an icon file and writes out a fragment of C code with the icon data structures. by Carolyn Scheppe          Program to merge the MemList entries of sequentially configured RAM boards, by Carolyn Scheppe          An object oriented drawing program, V1.1 by Tim Mooney</p> <p><b>Fred Fish Disk 57</b>          Replaced by FF97 Due to Copyright problems</p> <p><b>Fred Fish Disk 58</b>          ASDG-rd Extremely useful shareware recoverable ram disk, by Perry Kivolowitz          Displays any IFF picture, independent of the physical display size, using hardware scroll, by John Hodgson          Reads pairs of x and y value from a list of files and draws a formatted graph, by Laurence Turner          Shareware data management system, V1.5          Walks through the free memory lists, zeroing free memory along the way, by John Hodgson</p> <p><b>Fred Fish Disk 59</b>          NewZAP A third-generation multi-purpose file sector editing utility. V3.0 by John Hodgson          RainBow A Maunder-Style rainbow generator, by John Hodgson          SMUSPlayers Two SMUS plays, to play SMUS IFF music formatted files, by John Hodgson          View A tiny ILM viewer by John Hodgson          WBDump JX-80 optimized wordcount printer that does not use DumpRPort, by J. Hodgson</p> <p><b>Fred Fish Disk 60</b>          Browser Update to browser program on disks 18 and 34, S-E          Browser2 Another different browser program. E          Clock Clock program with fonts, colors. E          Dime Dillon text editor V1.22 for programmers,ED          DropCloth Puts pattern on Workbench windows,ED          DropShadow Puts shadows on Workbench windows,ED          FixWB Similar to DropCloth, but doesn't work yet, S-D          mCAD Object-oriented drawing program, version 1.2.2. Much improved over disk 56.          Robotroll Demo of animated pointers on Workbench, S-E-D          Supermort General compounding/amortization loan calculator. E-D</p> <p><b>Fred Fish Disk 61</b>          Various shareware and freeware programs          Blitz Memory resident file viewer. Very fast. E-D          BlitzFonts Makes text output faster. E-D          HandShake Terminal emulator with VT52/VT100/VT102support. E-D          Med Mouse-driven text editor version 2.1. E-D          PrtDrvGen Generates printer drivers, version 1.1.5 available from author. E-D          Show Slideshow-like IFF viewer, V2.1. E-D          Uedit Customizable text editor V2.0. E-D          Ueturbio Example Uedit setup macros. S-E-D</p> <p><b>Fred Fish Disk 62</b>          ATPatch Patches Transformer to work under AmigaDOS 1.2. S-E-D          FillDisk Writes zeroes to free blocks on a disk for security. S-E-D          LPatch Patch for programs that abort when loading under AmigaDOS 1.2. S-E-D          MicroEmacs Conroy MicroEmacs V3.8b, newer than disk 22. S-E-D          PearlFont Like Topaz, but rounded edges.          Terrain Generates fractal scenery. S-E-D          VSprites Makes 28 Vsprites, from P&amp;E Disk.</p> <p><b>Fred Fish Disk 63</b>          This is a port of the Unix game 'Hack', by the Software Distillery, version 1.0.30.</p> <p><b>Fred Fish Disk 64</b>          This is a port of the Unix game 'Lam', by the Software Distillery, version 12.0B.</p> <p><b>Fred Fish Disk 65</b>          This is an official IFF specification disk from Commodore, an update to disk 16.</p> <p><b>Fred Fish Disk 66</b>          Bank Unix text processor, like 'awk'. Doesn't work, but source is included. S-E-D          MWB Example of rerouting Workbench window open calls to another custom screen. Version 1.01. S-E-D          Example for closing a custom Workbench screen. S-E-D          Generates one-line fortune-cookie aphorisms. S-E-D          Build your own mouse port clock.          Creates C source files for menus, based on text descriptions. S-E-D          CBM tutorial on new packets and structures in AmigaDOS 1.2          PascalToC Pascal to C translator, not so great. S-E-D          'rator-like FORTRAN preprocessor. S-E-D          Starts programs from CLI, allowing CLI window to close. E-D</p>
--	--	---	---



SunMouse	This program automatically clicks in windows when the mouse is moved over them. V1.0, E-D	AutoIconOpen	Fools WB into thinking mouse has double-clicked icons. In C, S-E-D	ScalDisplay	Smush Target Adventure AmiTerm	hack created from "In" Smushes an IFF file. Each mouse click becomes a gunshot	Adler, and Warren Usui. ADL enhancements by Ross Cunniff. Included are sources to the ADL compiler, interpreter, and debugger. Binaries combined by Ross with Lattice 3.03. CLI environment only. Documentation is available from the authors.	
<b>Fred Fish Disk 66</b>		Dio	Generic Exec device interface code for opening libraries, getting multiple IO channels, asynchronous operations, etc. In C, S-E-D	<b>Fred Fish Disk 82</b>	Port of the classic Crowther and Woods game V0.50 of a telecommunications program, with scripts, radial, beeps, enhanced file requester	Amiga port by Joel Swank	<b>Fred Fish Disk 92</b>	portable 6502 assembler, C source, by J. Van Ornum, Amiga port by Joel Swank
AmScsi	Preliminary plans for a SCSI disk controller board.	Dissolve	Slowly displays IFF files, also Nov 86 Dr. Dobb's program. In C, S-E-D	D2D-Demo	Demo version of Disk-2-Disk by Central Coast Software		Bawk	Text processor update from FF65 inspired by UNIX awk. Searches files for patterns, performs actions based on patterns. By Bob Brodt; Amiga port by Johan Widen
Asm68k	Macro assembler, version 1.0.1. E-D	DTem	Re-arranges windows so that at least one pixel of menu bar gadgets are exposed. InC, S-E-D.	DX-Synth	Voce filter program for Yamaha DX central SEs synthesizers, update to disk 38		HunkPad	update of FF84 version, by J. Hamilton,pads an object file to a multiple of 128 bytes for better xmodem transfer. S-E
Assigned	Example for avoiding DOS insert-disk requester, by scanning the list of assigned names. S-E-D	Expose	Scans a text file, converts to C-style printable strings. C,v2.0, S-E-D	DiskMan	V1.0 of another DirUtil program		Less	Like Unix "more", better, version 1.2 update of FF74. Scrolls back and forward. S-E by Mark Nudelman, Amiga port by Bob Leivan.
Dk	Pretends to eat away at CLI window. S-E-D	Lit	"Long Movie", program views series of IFF pics in quick succession, up to 19s. Shareware, E-D	Icons	Miscellaneous new icons		Ndr	Library that implements the 4BSD Unix dir access routines by MikeMeyer. S
Flip	Flips whole screen as a joke. S-E-D	Lmv	Mouse pointer disappears after ten seconds of non-use. In C, S-E-D	Rocket	Another Workbench hack, plays Lunar Lander		Parse	Recursive descent expression parser, computes, and prints expressions. Includes transcendental function support. C source included, by J. Olsen
Fooool	Fooool cross-compiler generates VAX assembly code. S-E-D	MouseOff	Examples of controlling parallel port with resources instead of the PAR: device. In C,S-E-D	Sand	Game of sands following your pointer.		Shar	Two programs to pack and unpack shell archives includes C source, by Fabbian G. Duloe
Free	Prints amount of free space on all drives S-E-D	ParOut	Examples of controlling parallel port with resources instead of the PAR: device. In C,S-E-D	<b>Fred Fish Disk 83</b>	This disk contains a demo version of TeX from N Squared. It is limited to small files, and the previewer can only display ten pages or less, and only a small number of fonts are provided.		SmallLib	8 times smaller AmigaLib replacement, binary only. by Bryce Nesbitt
MailTest	Pretends to melt the screen. S-E-D	PenPalFont	Similar to RunBack on disk 66, runs program from the CLI allowing the CLI window to close. In C,S-E-D	<b>Fred Fish Disk 84</b>	AudioTools/Programs from Rob Peck's July/August Amiga World article		Uuencode	Encode/decode binary files for e-mail or text-only methods. Update of FF53, includes checksum technique, compatible with older versions, plus transparent to older versions options. By Mark Horton, modified by Alan Rosenblatt & Bryce Nesbitt.
Nart	Graphic flying string demo. S-E-D	RunBackGround	Screen dump utility,update FF 66-E-D	BitLib	Blitter experimentation program, V1.2, update to FF69		<b>Fred Fish Disk 93</b>	Version 1.27 WYSIWYG programmer editor. Not a word processor. Includes key mapping, fast scrolling, title-line statistics, multiple windows, ability to iconify windows. Update of FF87, SE by Matt Dillon
Purty	Easy way to set printer attributes from Workbench. E-D	Snapshot	Example installs a device handler before Intuition, and speaks each key as it is pressed. In C and assembler, S-E-D	Simple editor, similar to Unix 'ed', based on the editor in Software Tools.			MicroEmacs	Version 3.8, update to FF61 includes source. Orig by Dave Corry modifications by Daniel Lawrence
RayTracer	Simple ray tracing program. E-D	TypeAndTel	Prints info about system lists, in assembler,S-E-D	GravityWars	Game of planets, ships and black holes, v1.04, update to disk 70.		<b>Fred Fish Disk 94</b>	Demo programs from Rob Peck's July/August issue of AmigaWorld on accessing the audio device.
SendPackets	Updated CBM examples of packet routines on disk 35. S-E-D	Xplor	Edits and recalls CLI commands, v1.3, E-D	HunkPad	Adds legal padding to executables for Xmodem transmission.		ClickUpFront	V2 update of FF84. S. by Rob Peck
SnapShot	Memory resident screen dump. E-D	Cied	Intercepts graphic printer dump calls and accesses color map, width, and screen resolution. C, S-E-D	PipeHandler	An AmigaDOS pipe device which supports named pipes and taps. V1.2		HeliosMouse	Automatically activate a window simply by moving the mouse pointer into the window. V1.0. Includes source. By Davide Cervone
TagBBS	Shareware BBS system, version 1.02.	Control	Simple WYSIWYG text editor for programmers, v1.25. Update of FF59-E-D	PopCLI	V3.0 of a hot-key to invoke a CLI window, with screen blanker, update to disk 40.		IFF2Ps	Convert any IFF file to postscript for printing or viewing on a postscript compatible device. Version 1.2, by William Mason and Sam Paolucci E
<b>Fred Fish Disk 67</b>		Dme	WB dropshadows, v2.0 update FF59. E-D	Requester	Update FF34, file requester similar to DPaint.		ModulaTools	Various Modula 2 prog. routines, by Jerry Mack
AmCat	Shareware disk cataloging program.	Funds	AmigaBASIC prog tracks mutual or stocks-D	ScotDevice	V33.1 of a 'mountable MicroForge SCSI driver.		Terra3D	Pseudo-random 3D relief scenery generator, update of "sc", FF87, by Chris Gray, 3d by Howard Hull
AmigaSpell	Shareware Intuition spelling checker, V2.0. E-D	Less	Text viewing program, like Unix "more", v1.1, update to disk 34. S-E-D	Viacom	Another Schweb hack, makes TV-like static on screen.Parody		<b>Fred Fish Disk 95</b>	redirects the serial device or parallel device output to a file. Capture print jobs, debug or "offline" printing V4 By C. Scheppner S
Bouncer	3-D bouncing ball written in MultiForth, SED	MakeMake	Scans C source files and constructs a vanilla 'makefile' in the current directory. S-E-D	<b>Fred Fish Disk 85</b>	V2.06 of Dillon's 'tsh'-like shell Source to wildcard file requester		CygnusEd	Demo of CygnusSoft's CygnusEd editor, a multiplatform, multiple feature editor, includes demo 3.0 of MandFXP, by CygnusSoft Software E
Comm	Terminal program version 1.33. E-D	mCAD	Object-oriented drawing prog, v1.2.4, update to FF59 Shareware, E-D	Cah	Hides expansion memory from programs		Gomf	"Get Outta My Face" makes the Gnu go away to allow clean-up & shutdown more cleanly. V1.0, by Christian Johnson E
Dux5	Another version of DirUtil. S-E-D	Random	Simple random number generator in C. S-E-D	FileReq	Shareware tools to manipulation IFF images		Journal	records sequence of mouse & keyboard events, stored in a file. Future playback. Good for demos or documenting bugs. E by D. Cervone
HexCalc	Hex, octal, & decimal calculator. E-D	TDebug	Monitors devices by intercepting Exec Send() and DoIO() vectors, in C, v1.0, S-E-D	Hide	ServerShare library to add in low memory situations		MergeMem	attempts merging of MemList entries of sequentially configured ram boards. Allows allocating a section of memory which spans both boards. V2, update of FF56, by Carolyn Scheppner S
Icons	Various big and alternate image icons. Mandala graphics and sound. E	Units	Converts measurements in different units, includes "chart" option, in C, S-E-D	ImageTools	A star plotting program with source.		PrinterStealer	Asimilar to "Cmd", allows diversion of output destined for print to file. Binary only. Source avail. from authors. By A. Livshits & J-M Forgeas
Mandala	Mandala graphics and sound. E	XCopy	Replaces measurements in different units, includes "chart" option, in C, S-E-D	LowMem	Example of setting raw mode on standard input		Record-Replay	similar to "Journal", records and plays back mouse and keyboard events. B only. source avail. from authors, Alex Livshits & J-M Forgeas
PerfMat	Demo shareware personal file manager.	<b>Fred Fish Disk 75</b>		Plo6	Lunar Lander for Workbench, with source.		<b>Fred Fish Disk 96</b>	Animation reader and display by the combined efforts of Videocase, Sculp3D, Silver, Forms-In-Flight, and AnimatorAppletree M Hashel.
RSLClock	Menu bar clock version 1.3. E-D	Bezier	Play with Bezier curves points and granularity, S-E-D	RawIO	Example of setting raw mode on standard input		Chess	Amiga port, non-Amiga interface. High playability. V1.0. S. by J. Stanback, Amiga port by B. Leivan
RTCCubes	Graphics demo of 3D cubes. E-D	BSplines	Play with B-splines, as above, S-E-D	Rocket	Update to disk 75 of Unix-like 'diff', S-E-D		Hackbench	provides source for WB-Likeprog, for experimentation & validation of new interface ideas. Not a WB replacement. by Bill Kinnersley
Wheel	"Wheel of Fortune"-type game,AmigaBASIC	Copy	C source for Comm terminal program v1.34. S-E-D	Vmore	V2.27 of Dillon's text editor, update FF74-E-D		Label	Print labels with arbitrary text. V1.3. Source available from author, M-Hansen
<b>Fred Fish Disk 68</b>		Diff	Simple 'diff' in C, S-E-D	Vnews	Simple Unix news reader.		LineDrawer	Produces line drawings based on drawing commands stored in a text file. Includes demo that draws an outline map of the USA and state borders. V1.0, SE. by John Olsen
This is version MG 1b of the MicroGnuEmacs. Source and executable are included, as well as source for other computers besides the Amiga.		DulM2	Another DirUtil in Modula-2, v1.5, S-E-D	<b>Fred Fish Disk 86</b>	AutoPointAuto-selects window under the mouse pointer, with screensaver.		PopUpMenu	Example code implementing pop-up menus, reasonably compatible with Intuition menus SE. by Derek Zahn
<b>Fred Fish Disk 69</b>		Eless	Fast 'dir' program in C, S-E-D	ClickToFront	Double-clicks on window brings it to front, v1.1, S-E-D		Tek4695	Tektronix 4695/4696 printer driver. SE. by P. Staub
Asm68k	Macro assembler, v1.0.3, E-D	Fd	Faster 'less' in C, S-E-D	Cmd	V3.0 of a tool to redirect printer output to a file.		TimeRam	Fast text rendering routine, SE. by B. Takahashi
BitLib	Blitter exploring program, in C, S-E-D	HardCopy	Sends a transcript of a CLI session to a file, in C, S-E-D	FileISG-Demo	Demo of Softwood File ISG, a database manager with sound and graphics.		WarpText	Fast text rendering routine, SE. by B. Takahashi
Conman	Replacement console device handler adds editing and history to any application that uses CON: v0.9, E-D	<b>Fred Fish Disk 76 &amp; 77</b>		Install	Alternate AmigaDOS 'install' programs,SED		<b>Fred Fish Disk 97</b>	Implementations of Unix cut and paste commands. by John Weald
Console	Replacement console routines, in C, S-E-D	These are disks 1 and 2 of Chris Gray's Draco distribution for the Amiga. Draco is a compiled, structured language reminiscent of both C and Pascal. A full interface to AmigaDOS and Intuition is supplied. Be sure to get both disk 76 and 77.		AdvSys	Adventure system from Byte May 1987, v1.2 E-D		Graphit	Program to plot simple functions in 2 or 3 dimensions. by Flynn Fishman
Dk	Replaces the screen bit by bit, update to disk 66, in Modula-2, S-E-D	<b>Fred Fish Disk 78</b>		AutoIconOpen	Fools Workbench to open disk icons, V1.2		Juggler	V1.2 of robot juggler animation. Uses HAM mode and ray tracing, by Eric Graham
Frag	Displays memory fragmentation by listing the size of free memory blocks, in C, S-E-D	Cycles	Cycle game like "Tron", v1.0, E-D	Ciaz	Converts IFF files to PostScript, V2.0, SED		MouseReader	Shareware program to read text files & view IFF files using only the mouse, by William Bette
IconType	Change the type of an icon, in C, S-E-D	EOIMS	Experts Only Mercenary Simulator game, E-D	Commidi	SetMackraz's Commodities Exchange, an exec library to manage input handler, v0.4		Spines	Prog to demonstrate curve fitting & ren-derr techniques, by Helene (Lee) Taran
Make	"make" in Marx C, S-E-D	Mandelbrot	Mandelbrot generator with enhanced palette controls, fixed/float point, presets, v1.50, in Marx C, S-E-D	Diff	Update to disk 75 of Unix-like 'diff', S-E-D		Shm	Graphics demo, approximately simulates the motion of two interacting pendulums. Includes S by Chris Ediss
MonProc	Monitors processes for packet activity, in C, S-E-D	MandelVroom	Mandelbrot generator with enhanced palette controls, fixed/float point, presets, v1.50, in Marx C, S-E-D	Dme	V1.27 of Dillon's text editor, update FF74-E-D			
MouseClock	Mouse pointer into a digital clock, in C,SED	<b>Fred Fish Disk 79</b>		DropShadow	V2.27 of Dillon's text editor, update FF74-E-D			
Sb	Browses system structures, from Transactor magazine, v1.0, in C, S-E-D	AsmTools	CLI tools in assembler: echo, loadit, mounted, setsize, why, S-E-D	Elib	An AmigaDOS device handler generates unique identifiers, V1.0, S-E-D			
Spew	Generates "National Enquirer"-type headlines from rules file, in C, S-E-D	AssignDev	Gives devices multiple names, in C, S-E-D	ID-Handler	Alternate AmigaDOS 'install' programs,SED			
Spool	Three programs to demonstrate multitasking & spooling in a printer spooler. In C, v1.2, S-E-D	AuxHandler	Example of a dos handler that allows use of a CLI via the serial port. Includes source. Author: Steve Drew	Install	Alternate AmigaDOS 'install' programs,SED			
Wc	Counts words ala Unix 'wc', but faster, in C, S-E-D	Cmd	Redirects printer output to a file, in C, S-E-D	MemWatch	Waits for low memory trading. V2.0, SED			
<b>Fred Fish Disk 70</b>		Info	AmigaDOS 'info' replacement, in C and assembler, S-E-D	MovePoint	Moves pointer to given location, S-E-D			
This is a disk of shareware programs.		Kill	Removes a task and its resources, in C, S-E-D	MoveWindow	Moves window to given location, S-E-D			
AmigaMonitor	Explores state of the system, v1.13	M2Error	Displays errors from TDI Modula-2 compiles, S-E-D	MunchingSq	Munching Squares hack, S-E-D			
Arc	Standard file compressor and librarian, v0.23, a port of MS-DOS v5.0. E-D	MonProc	Update to process packet prog. from FF69 in C, S-E-D	PaTest	Test to see if this is a PAL machine, S-E-D			
BlackBook	Phone book program.	Mounted	Program for testing if a drive is present, in a script in C, S-E-D	Sc	Generates random scenery, S-E-D			
DoTri	Intuition-driven file manipulator program, v2.0.	Nro	Another 'roll'-style text formatter, in C, S-E-D	Tek4695	Tek4695 printer driver			
GravityWars	Game of planets, ships and black holes, v1.03.	ParTask	Finds parent task, in C, S-E-D	WduallPF	Example of dual-playfield screen, update FF41, S-E-D			
Jobs	Alternate user interface to CLI and WB, v2.1.	QueryAny	For scripts, asks a question, accepts Y/N, gives return code. In assembler, S-E-D	WarpText	Fast text rendering routines, S-E-D			
Lens	Magnifies area around mouse, shows it in a window, v1.0.	ScnSizer	Resets pref settings for screen size, in C, SED	Yaff	Example IFF reader, S-E-D			
Life-3d	3D version of the classic cellular-automaton game, v1.2.	SharedLib	Example, shared lib, in C & assembler, S-E-D	Zoo	A file archiver like 'tar', v1.42A, E-D			
Logo	Logo language interpreter	Task	Simple CreateTask() example in C, S-E-D	<b>Fred Fish Disk 88</b>	(see Fred Fish 89)			
SetKey	Demo keypad editor, v1.0	Uw	Unix Windows client v1.0, in C, S-E-D	FF Disk 88 has been removed due to copyright problems				
Vpg	Makes displays for aligning video monitors, v1.0.	Who	Lists tasks on ready and wait queues, in C, S-E-D	<b>Fred Fish Disk 89</b>	(replaces Fred Fish 80)			
<b>Fred Fish Disk 71</b>		<b>Fred Fish Disk 80</b>		Disk catalogue program, V1.0a, E-D				
AirFoil	Makes airfoils using the Joukowski transformation, in C, S-E-D	Fred Fish 80 has been withdrawn due to copyright problems.		Shareware function key editor, V1.01, E-D				
Amiga Basic	Miscellaneous programs including 3D plot program, a kaleidoscope, C-A logo drawing program, file comparison utility string search program, S-E-D	<b>Fred Fish Disk 81</b>		Demo of MicroFie File database prog				
Books	A variation of "lines", but with variable color blocks. E-D	Asm68k	V1.1.0 of a macro assembler	Adjust screen position ie Preferences,SED				
Comm	Great terminal program, v1.34, E-D	AutoFac	Shrinks the FACD window and moves it to the back	Bouncing squiggly lines demo, S-E-D				
DiffX	Utility for exploring file system E-D	Brushes	53 custom IFF brushes of electronic symbols	screen contraction requester improvement S-E-D				
Fpic	Simple image processing program that operates on IFF pictures, with several filters, merging images, E-D	CheckIFF	Checks structure of an IFF file CledV1.4 update FF74 of a simple CLI	Display Hack S-E-D				
IconMk	Makes icons for files, v1.2a, E-D	Conman	Replaces console handler to add editing and history to many programs	<b>Fred Fish Disk 90</b>	(replaces Fred Fish 80)			
NewFonts	Two new fonts: 'shalt18', an electronic circuit element font, and 'lrm5', a PC-like font.	Fonts	Miscellaneous fonts	AmiGazer	Night sky view of 1573 stars, set date, time, day. E-D			
PetCli	An AmigaBASIC CLI shell program.	Icon	V6.0 of the icon programming language	CardFile	AmigaBasic card file study aid. E-D			
PWDemo	Demo of the commercial product PowerWindows, v1.2. It aids creation of custom windows, menus, and gadgets, giving C or assembly source. E-D	KeyLock	Freezes the keyboard and mouse until pass word entered.	Conman	Console handler replacement gives line editing and history to most progs, v0.98 E-D			
Rot	Creates and animates 3-D objects, v0.5, E-D			IMandelVroom	Slight update to disk 78 Mandelbrot program, E-D			
TimeSet	Sets time from Workbench, E-D			NewDemos	Replacements for lines and boxes demos that take less CPU time, E-D			
<b>Fred Fish Disk 72</b>				Othello	Game of Othello, E-D			
This is a disk of IFF pictures.				PrintText	Displays text files with gadgets, speech, IFF display, v1.2, E-D			
<b>Fred Fish Disk 73</b>				PrtDrvGen	Automatic printer drv. generator, v2.2b,ED			
Add	Customizes existing program menus with Amiga-key shortcuts. Also includes 'unif', which waits until a given window is created. Shareware, in C, S-E-D			RainBench	Cycles colors of WB backdrop or text. ED			
				ShortOut	Makes single-key shortcuts for entering commonly typed CLI commands.& custom macros. E-D			
				ShowPrint	Displays and prints all sizes of IFF pictures & controls printer output styles, v2.0 E-D			
				Sizzlers	Graphics demos, v1.7.0, E-D			
				Timer	Small Workbench timer counts time and \$/ minute. E-D			
				<b>Fred Fish Disk 91</b>	Adventure Definition Language (ADL) a superset of an older language called DDL by Michael Urban, Chris Kostanick, Michael Stein, Bruce			



<b>Fred Fish Disk 98</b>	Access	16 color terminal program based on Comm V1.34. Includes Macro window, custom gadgets, colorized menus, etc. V. Beta 0.18 by Keith Young .com by D.J. James. E.
Backup	Writes AmigaDOS disks as the backup destination. recovers files from the backup disk. Requires manual decisions on disk structure. by Alan Kent SED	
DCDemo	DishCart 2.3, a disk catalog program, demo limited to cataloging 100 files at a time. by Ed Alford, MicroAde Software	
HdDriver	WD-1002-05 hard disk controller driver. Card capable of maintaining 3 hard disks and 4 floppies, the driver is capable of only one hard disk. by Alan Kent SED	
QBase	Quick-Base, a "MailBase Management utility", define and maintain a maximum of 200 records per file. by Kevin Hamise E	
Thai	Thai language quiz program. Speak or type english/Thai sentences from supplied file. by Alan Kent SE	
<b>Fred Fish Disk 99</b>	A-Render Version 3.1 Ray-Tracing Construction Set for the Amiga Computer by Brian Reed ED	
Berserk	Must see animation, by Leo Schwab	
Comman	Console handler replacement, provides line editing and command line histories transparent to application prog uses CON: windows. Shareware V1.0 by W. Hawes. E.	
WBLander	Workbench display hack game, upgrade of "Rocket" on FF85, now with sound effects. By Peter da Silva. E	
<b>Fred Fish Disk 101</b>	CrPlane	Circular plane generator for VideoScape3D. Generates a clockwise circular polygon with the specified number of vertices. V1.0 by T. Florian SE
IconAssembler	Change Workbench icons with IFF-brush files by Stefan Lindahl E	
Microspell	Standalone spelling checker scans text files and reports errors. 1000 common word list, 43,000 word main dictionary with multiple user dictionary support. Interfaces with MicroEMACS 3.9 with an emacs macro to step through the source file, stopping at suspect words and allowing the user to option. V1.0 by Daniel Lawrence, SED	
Midi	Midl library and utility set. Includes Midl monitor, routing utility, status utility, and more. by Bill Barton SED	
Psintp	Postscript Interpreter reads and previews files on screen. by Greg Lee S(easy)E	
Startups	Three C startup file replacements for standard Asitup.obj and LStartup.obj. Options include (1) BothStartup.obj, for the Workbench programs or CLI programs with or without command line parameters. (2) WBSStartup.obj, for Workbench programs or CLI programs that require no command line parameters. (3) CLISStartup.obj for CLI programs that require command line parameters but do not need to be Workbench runnable. by Bryce Nesbitt SE	
<b>Fred Fish Disk 102</b>	Doug	Machine independent macro based C de-bugging package. Update FF41, by F. Fish prototyping support by Brinayk Banerjee SE
Match-stuff	Heavy duty text pattern matching stuff. Includes simple match text replacement capability. by Pete Goodeve	
Sectormax	Recover lost or damaged data from floppy or hard disks or repair a damaged volume. by David Joiner E	
SilCon	Smart input line interpreter with window for full editing. Upgrade FF50 by P. Goodeve, E	
Xicon	Use icons to call up scripts containing CLI commands. V2.0 upgrade of FF31 by Pete Goodeve E	
<b>Fred Fish Disk 103</b>	AviTrees	Library and test prog. implement routines for creating and using trees held in memory. S.
Calc	A programmable RPN calculator.	
Cref	A C cross ref. prog. S.	
Doskawk	A pair of progs. allows you to save files to one or more floppies for quick loading. Doesn't store Dos format.	
IntuDocs	A prog. to improve control and handling of the material on all disks in "CLI-area".	
MPF-Update	A text import util. for MicroFiche Filer (demo on FF 89) and updates to some PD disk library databases.	
Pack-it	Takes all files the files and dirs. on a disk & packs them into a single file, for modems.	
Sol	Amiga version of solitaire.	
<b>Fred Fish Disk 104</b>	AnalyticCalc	Is a large and powerful spreadsheet prog.
<b>Fred Fish Disk 105</b>	AsmProgs	Misc. assembly tools. Includes some S.
BasicProgs	LeastSquare solves least square probs. graphs results, S.	
Bison	A replacement for unix "yacc" command. S.	
Dmouze	Another prog in the tradition of display hacks". S.	
FlamKey	Allows keyboard and mouse inputs to be looked until a password is entered.	
GravityWars	Game of planets, ships & black holes. V2.0 FF84 update.	
IPo2C	A util. to write a C-lang definition to mimic the intuition pointer. S.	
Pret-er-Fill	Ex. of creating & using reentrant processes. S.	
Record-Reply	Similar to "Journal" on FF21, update to FF95.	
<b>Fred Fish Disk 106</b>	Furkey	Shareware function key editor, v1.1, update to FF89. Source avail. from author(Anson Mah).
MoreArt	A small selection of some Amiga artwork.	
QuickFix	An IFF slideshow and cel animation prog.v0.13.	
RistiNola	A Finnish game. Also called Go-Moku. v1.0	
<b>Fred Fish Disk 107</b>	Csh	V2.0? of Matt Dillon's csh like shell. S.
Diff	A util., similar to other common "diff" programs. S.	
ProSuite	Provides ex. code of facilities such as FileIO Requester, XText, DoRequest, & tutorial on how to program the Amiga. Book 1.01.S.	
SVTools	Some useful tools. S.	
<b>Fred Fish Disk 108</b>	AList	Dir listing prog. based on LD4 prg S.
DirMaster	Disk cataloger, v1.0b, update to FF89. S.	
Dots-Perfect	Printer Driver for an Epson MX80 printer with upgrade kit installed. S.	
MoniDCMP	Lets you monitor the IntuMessages that pass through an IDCMP window. Prints the message class, mouse coordinates, qualifier values. Great for debugging. S.	
PrintPop	A util. to send common control settings to PRINT: S.	
Sectormax	Utilities to recover lost or damaged data from floppy & hard disks. v1.1, an update to FF102.	
Tek	V100 emulator for a Tektronix 4010/4014. (V2.0) update to FF52. S.	
Zoo	File archiver, like "arc". v1.24B, update to FF87	
<b>Fred Fish Disk 109</b>	Machine	A new animation.
SimCPM	A CPM/M simulator 6809 along with h19 emulation. S.	
UUPC	Hook up your Amiga as a usernet node. S.	
<b>Fred Fish Disk 110</b>	A68k	A 68000 assembler written in C. S.
Pdc	An optimizing C compiler for the 68000 processor. update to FF53, but not based on that code.	
<b>Fred Fish Disk 111</b>	AmiLoad	A graphical monitor of cpu, blitter, & memory use. Includes two components: load.device, monitors system parameters, & amiLoad, which is the user interface & display program. by Jeff Kelley SE
AssignDev	Assigns multiple names to a given device. modified version of the original released on disk number 79. By Philip Lindsay, mod by Olaf Seibert SE	
Gauge	Continuously displays memory usage in a vertical bar graph Binary only. By Peter da Silva	
HeliosMouse	Another "summouse" prog. Automatically activates a window by mouse pointer V1.1, update to FF94. By David Cervone SE	
Labels	Alphabetic & numeric ordered cross reference lists of defined system constants. Recommended for debugging purposes only, use the symbolic values in prog! By Olaf Seibert	
Mandel	Mandelrot generator program, with bits & pieces of code from C. Heath & R.J. Mical. By Olaf Seibert S	
PopFile	A POPCUI type that plays like all over your screen. Lots of bits & pieces from Tomas Rokicki's blitab & John Toebes' POPCUI. By Olaf Seibert S	
<b>Fred Fish Disk 112</b>	BeachBirds	Beach scene portrayed by sprites & sound 512K machine. By Jerrold Tinnell B only
Bully	Pushes all open screens around (thus the name "bully"). Show more than one demo at a time By Mike Meyer S	
DropShadow	Dropshadow V2.0, use with Bryce Nesbitt's Wavebench demo. B only. By Jim Mackay	
HagenDemos	"RGB" & "Focus". RGB requires one meg. B only. By Joel Hagen	
Viacom	Latest version of viacom for use in conjunction with Wavebench demo. B only. By Leo Schwab & Bryce Nesbitt	
WaveBench	A neat screen hack, & runs on 512K machines. For more laughs, try in conjunction with Viacom or Dis (Dropshadow). Includes S. By Bryce Nesbitt	
<b>Fred Fish Disk 113</b>	AmiCron	Simple Unix "cron" type program, a background task uses a disk-resident table to automatically run certain tasks on a regular basis, at specific times. V2.3. S. By Steve Sampson, Amiga port by Rick Schaeffer V1.28 of Matt's text editor. A simple WYSIWYG editor for programmers. Not a WYSIWYG word processor Features: arbitrary key mapping, fast scrolling, line-line statistics multiple windows, iconify windows, etc. Update to FF93. S. By Matt Dillon
DosDev	Example DOS device driver in Marit C. Version 1.10, includes S. By Matt Dillon	
M2Amiga	Demo of M2Amiga. A fast single pass Module-2 compiler with editor, linker, a small set of interface & standard libraries. Compiles only small demo programs by limiting descriptors & imports. Further development of the ETHC compiler on FF24. B only. Demos with Source. By R. Degert, C. Needer, M. Schaub, J. Straube (AMS-Soft)	
NoIconFos	Clears position info of any icons, allows Workbench to pick a new place for the icon. Useful for disk & drawer icons where Snapshot rewrites the icon & the window information. Module 2, and/or demo for M2Amiga By Markus. Schaub	
<b>Fred Fish Disk 114</b>	COed	English to C (and vice versa) translator for C declarations, a must for anyone except possibly the most hardcore C guru. By Graham Ross, S.
Vt100	V2.7 of vt100 terminal emulator with kernel & xmodem file transfer. Includes a few bug fixes pointed to Usernet shortly after the posting of v2.7. Update to FF55. Includes S. By Dave Wacker	
WBLander	A special version of the WBLander program from FF100. Ending is unique. Effective use of sound. Includes S. By Peter da Silva & Karl Lehnbauer	
<b>Fred Fish Disk 115</b>	Killer	Massive Video commercial of the Amiga, Beatles music, requires one meg of memory to run. Binary only. By R. Wilt
Marketrod	Another devilsprite oriented demo with lots of "int" jokes. 512K required, includes S. By Leo Schwab	
<b>Fred Fish Disk 116</b>	Movies	A ram animation system with three different example animations: Kahnabanks, Rocker, & F-15. Kahnabanks & Hooker run on a 512K Amiga & show off overboard HAM mode. Includes a animation player program (movie), animation builder programs (dibm, plbm), & a textgraphics display program (wibm). By Eric Graham & Ken Otter
<b>Fred Fish Disk 117</b>	AMUC_Demo	A really neat horizontal scrolling demo that is a 2400 x 200 pixel 32 color IFF picture composed of digitized snapshots of members of the Amiga Users of Calgary, superimposed on a very wide picture of the Calgary Skyline. B only. By Stephen Vermuelen & Stephen Jean
Exp_Demo	Demo version of Express Paint 1.1. Used to create the scrolling demo picture in the AMUC_Demo drawer on this disk. B only. By Stephen Vermuelen	
<b>Fred Fish Disk 118</b>	Empire	Complete rewrite, in Draco, of Peter Langston's Empire. A multiplayer game of exploration, economics, war, etc, can last months. Use local keyboard or modem. V1.0, shareware, & S. By Chris Gray, original game by Peter Langston
RAAnimm	Displays lines whose end points are bouncing around the screen, which is a double buffered HAM screen. The Y positions of the points are continuously copied into an audio waveform and played on all four channels, & the pitch of a just intoned chord is derived from the average X position of these points. JForch. Source By Phil Burk	
Stars	Based on original code by Leo Schwab, credits longer than actual demo. Runs on 512K Amiga. B only. By Noble Oms	
WireDemo	Demonstrates the Amiga's line drawing speed. Runs on a 512K Amiga. Includes S. By Matt Dillon	
<b>Fred Fish Disk 119</b>	MicroEMACS	V3.9e of Daniel Lawrence's variant of Dave Conroy's microemacs. Update to FF93. Also included, for the first time, is extensive documentation in machine readable form. SE. Author: Dave Conroy, Enhanced by Daniel Lawrence
<b>Fred Fish Disk 120</b>	Amibuda	Clone of Space Invaders, one of the PDS games for the Amiga. B only B.Y. LateNight Developments
BackGarden	Graphical Background (an undergraduate A.I. course project). Version 1.0. S. By: Robert Pister	
Bankn	A complete checkbook system offered by the author as shareware. Version 1.3, binary only. By: Hal Carter	
EgyptianRun	"Road race" & "hazards" type game. Version 1.1, B only, shareware, source available from author. By: Chris Hames	
IconImage	Replace an old icon image with a new image, without affecting icon type, drawer data, etc. SE. By: Denis Green	
<b>Fred Fish Disk 121</b>	BasicStrip	AmigaBASIC prog. helps to convert programs written in other forms of Basic to AmigaBASIC. By: George Trepal
DataPlot	Shareware AmigaBASIC, plotting program. Also includes a least squares curve fit program. By: Dale Holt	
Plot	Shareware 3-D AmigaBASIC graphing prog. & sample output plots. Source available via author. By: George Trepal	
Stairs	AmigaBASIC prog. demos a musical illusion based upon perceptual probability of widely spaced tones whose volumes are defined as a sinusoidal relationship to their frequency. By: Gary Cuba	
Uedit	V2.3 of this nice shareware editor. With learn mode, command language, menu customization, and other user configurability and customizability features. Binary only, shareware, update FF86. By: Rick Stiles	
WBColors	Prog to change Workbench colors for progs that expect to be booted off their distribution disk but are run from a hard disk. SE. Author: Stefan Lindahl	
<b>Fred Fish Disk 122</b>	Asteroids	Asteroid game. The images and sounds are replaceable by the end user. Anything goes! By: Rico Mariani
ItzPcs	Interactive puzzle prog. takes any IFF file with up to 16 colors, and breaks it up into squares to make a puzzle which the user can then piece together. V1.0.S. By: Al Otter	
Names	A shareware program to create and manage mailing lists. Binary only. By: Ernie Nelson	
Pr	Utility to print listings in different formats. Similar to the Unix "pr" program. Includes source By: Samuel Pastorek	
PushOver	Board strategy game, AmigaBASIC. Push your pieces onto the board until you're in a row in any direction. S. By: R.Yost	
PuzzlePro	Create a puzzle from an IFF picture, which the user can then piece back together again. AmigaBASIC. V1.0, B only, shareware, source available from author. By: Syd Bolton	
<b>Fred Fish Disk 123</b>	Ap	AP stands for "AmigaDOS Replacement Project". Ap is an effort led by Charlie Field of Microplants Inc., to replace the current DOS in a compatible fashion, so that current programs will continue to work. Ap also makes whatever improvements are possible, so that current and future programs will work better. Various authors contributed work. One of Allen's entries to the Badge Killer Demo Contest. It amply is an inside joke relating to a well known Amiga's experience with a certain high-end graphics hardware manufacturer. Author: Allen Hastings
<b>Fred Fish Disk 124</b>	Icons	Some sample animated icons. By: L. Plost
Target	AmigaBASIC Nice graphic of target cards. Author: L.Plost	
<b>Fred Fish Disk 125</b>	Eliauts	Animation entry to the BK D Contest. Background music arrangement, requires Sonix to use. By: Kevin Sullivan
<b>Fred Fish Disk 126</b>	Color	Manipulate the colors of specific named screens, saving current color sets to data files, loading new color sets from data files, or interactively changing colors. S. By: J. Russell
Dance	Two programs, "dancing polygons", are entries to the BK D Contest. They are similar, but demonstrate the range of colors available on the Amiga. S. By: John Olson	
Horribl	Animation entry to the BK D Contest. First known animation using the "Extra Hall Entry" mode. By: Kevin Sullivan	
Iconify	Subroutine creates an icon on the Amiga screen that can be subsequently dragged around, and double-clicked on. You can use this to have your programs "iconify" themselves to temporarily get out of the user's way. With source & demo program. By: Leo Schwab	
OnlyAmiga	Animation entry to BK D Contest. Three balls being juggled by pyramids rotating on their tops. By: Kpal Singh Hans	
Supib	Support library needed to rebuild various programs of Matt's from source, including DME, DTERM, etc. S. By: Matt Dillon	
Vcrack	V1.2 of virus detection prog. from Commodore Amiga Technical Support. Will test for the presence of a virus in memory, or on specific disks. B only. By: Bill Koester	
<b>Fred Fish Disk 127</b>	Bounce	Entry for BK D Contest. Creates little dots that bounce around and multiply. S. By: Steve Hansel and Tom Hansel
Nunesa	Entry to BK D Contest. It is quite small for what it does, and won't place in the contest. B only. By: Mark Riley	
Hippies	Entries to BK D Contest. Unlike most other animations, it shows a fixed object from a moving point of view, instead of vice versa. By: Allen Hastings	
<b>Fred Fish Disk 128</b>	Dir	68000 disassembler, written in assembler. S. By: Greg Lee
DropCloth	Place a pattern, a 2 bipane IFF image or a combination of a pattern and image, into the Workbench backdrop. Version 2.2, shareware. B. By: Eric Lawlisky	
LedClock	An extremely simple clock program, for interfaced screens only. S. By: Al Otter	
MRBackup	Hard disk backup utility. Does a file by file copy on AmigaDOS floppy disks. With an intuition interface & file compression. V1.3. Source. By: Mark Hinfel	
Paint	Simple screen painting program, written in web. Requires web preprocessing program to rebuild from source. Includes source in web. Author: Greg Lee	
PrDriver	A printer driver for the Toshiba "3 in one" printer in its Cume (best) mode. Includes source in C and assembler. By: Rico Mariani	
SOBackup	A hard disk backup utility. CLI interface only. Does file compression. V1.1, binary only. By: Steve Drew	
Sed	A clone of the Unix sed (Stream Editor) program. Includes source. By: Eric Raymond	
Keys	"Hot-keys" program binds keyboard function keys to window manipulation functions (window activation, front to back, moving screens, etc). S. By: Davide Cervone	
<b>Fred Fish Disk 129</b>	Doskawk	A pair of programs which allow you to save files, or a group of files, to one or more floppies for quick loading. Does not store files in DOS format, for speed. V2.0, update to FF103. B. Shareware. By: Gary Kemper
MRBackup	A hard disk backup utility, does a file by file copy to standard AmigaDOS floppy disks. Includes intuition interface & file compression. V2.0 (with sources) and 2.1 (binary only, source available from author). Update of FF128, by Mark Hinfel	
PaintJet	HP PaintJet printer driver. from HP sources.	
Patn	Two independent ports of Unix utility "patch", which applies context diff's to text file to automatically update them. Patch V1.3 was ported to the Amiga by Rick Coupland and patchV2.0 was ported by Johan Widen. S. By: Larry Wall	
<b>Fred Fish Disk 130</b>	DirMaster	Shareware disk cataloger, V1.1, update of FF108, new features and enhancements. B only. By: Greg Peters
Evo	Human evolution bytutrial with source. By: S. Bonner	
Hp	RPN calculator prog. supports calculations with binary, octal, decimal, hex, float, & complex numbers. Includes 32 registers for storing data & transcendental functions. V1.0. S. By: Steve Bonner	
Mach	"mouse accelerator" prog. with hotkeys, features of sun mouse, clickfront, and populi, a tile bar clock with a bos online charge accumulator, etc V1.6a, S. By: Brian Moine	
PatEdit	A pattern editor for creating patterns to input to the Amiga SetAPI macro call. Call sets the area fill pattern for the area filling graphics (Hectfill, AreaDraw, etc). Includes source. By: Don Hyde	
QMan	Mandelbrot generator written partially in assembler. For speed. Includes source. By: Steve Bonner	
<b>Fred Fish Disk 131</b>	Dic	Copies disks like Maudrader, but multitasks. Replaces diskcopy and format (smaller than either). Intuition interface. S. By: Tomas Rokicki
HyperBase	Shareware database management system. V1.6, Binary only, source available from authors. FF58 update. By: Michael MacKenzie, Marc Mengel, & Craig Norberg	
Life	A new version of Tomas's ancient Life game, with a new macro language for setting up patterns, good examples. S. By: Tomas Rokicki	
Mackie	A Poplil replacement that draws pretty lines on the screen in blanking mode. Includes source. Author: Software Division; enhancements by Tomas Rokicki	
Mgib	A version of Mgib with an Amiga port and other improvements by Tomas Rokicki. Define macros & bind them to function keys in startup file. Includes source. Author: Various; enhancements by Rokicki	
Wfrags	Another version of Frags. Pops up a little window that updates occasionally. Good for developers to monitor what progs. are doing to memory. S. By: Tomas Rokicki	
<b>Fred Fish Disk 132</b>	Bounce	Animation, a "must see" for every Amiga user, and ranks with "Juggler" as a premier demo for the Amiga. The difference between this distribution, and FF100, this one includes "source", use it as an example for creating animations. Fred Fish felt it was appropriate to have at least one animation that was available at the "source code" level. Author: Leo Schwab
<b>Fred Fish Disk 133</b>	Comman	Shareware replacement for the standard console handler, provides line editing and command line histories completely transparent to any application program that uses CON: windows. V1.1, binary only, update FF100. New features include additional editing keys, fast search keys, undo key, clear history command, and more. Author: William Hears
Crc	Two programs useful for generating 16-bit CRC listings of the contents of disks, and verifying that a given disk's files still compute to the same CRC's as listed. V1.0, binary only. By: Don Kindred	
CrcLists	Complete CRC check files for FF1128 using the Crc program included on this disk. These were made directly from Fred's master disks. Author: Fred Fish	
OverScan	Fixes the intuition library so that sizable windows with MaxHeight of 200 (400 in interface) and screens with Height of 200 (400 in interface) will take advantage of the PAL overscan capability of intuition V1.2. Useful only for European users who wish to run software written for the US market, without modifying the applications, but still using the additional space. S. By: Al Freund	
<b>Fred Fish Disk 134</b>	BongThrows	50 frame HAM animation done with Sculpt-3D, and DigPaint. The animation took about 325 hours of time to generate. By: Marvin Landis
Browser	Workbench tool, using text-only windows, makes all files in the system accessible for viewing, copying, moving, renaming, deleting, etc. Billed as a "programmer's workbench". V1.2, binary only. By Peter da Silva V1.29 of Matt's text editor. Simple WYSIWYG editor designed for programmers. Arbitrary key mapping, fast scrolling, line-line statistics multiple windows, & ability to iconify windows. FF113 update. S. By: Matt Dillon	



Find	Utility searches for files that satisfy a given boolean expression of attributes, starting from a root pathname and searching recursively down through the hierarchy of the file system. Like the Unix find program. V1.0, includes source. By: Rodney Lewis	RemLib	Gotz Muller Removes a specified library (if currently unused) or displays some info on all available libraries. Source in assembler. By: Helko Rath	Fred Fish Disk 148	EFJ "Escape from Jovii" A machine-code game featuring hi-res scrolling, large playfield, disk-based Hi-Score list, stereo sound, and multiple levels. Use a joystick in port 2 to control the ship. B. shareware (\$8). By: Oliver Wagner	WYSIWYG word processor in the traditional sense. Features include arbitrary key mapping, fast scrolling, title-line statistics, multiple windows, and ability to iconify windows. Update to version on disk number 134, includes source. By: Matt Dillon	
Library	Demo version of a shareware program that stores textual information without regard to structure or content, and allows complicated searching for specific patterns. B only. By: Bill Brownson	TurboBackup	A fast mass floppy disk duplicator with enforced verify mode to prevent errors. V1.0, binary only. By: Steffen Stempel and Martin Kopp	Fme	Nicely done map editor for the Fire-Power (tm) game. Features interlaced hi-res with intuition interface. See the "Readme.txt" file for information on making a bootable disk. Includes source. Author: Gregory Mackay	HP11	Emulates an HP11C calculator including the program mode. Features an ON/OFF button that turns the calculator into an icon that will sit and wait until you need it again. Documentation on the features is scarce, perhaps some industrious HP owner could write a small tutorial for the benefit of those that don't own an HP calculator. Binary only. Author: David Gay
SmartIcon	Shareware Intuition objects iconifier. V1.0 is limited to iconifying windows, adds a new "iconify gadget" to each window, when clicked, iconifies the window into an icon in the ram; disk. B only, source available from author. By: Gauthier Groult	WARanger	Sends a window, identified by its name, to the front or back, without selecting it. Useful with AmiCrn. Works on all screens. Includes source in assembler. By: Helko Rath	HandyIcons	Adds a menustring to the WorkBench window that allows you to run selected Workbench Tools by menu selection. Can be set up to provide custom environments. Current version supports only WorkBench Tools and not Projects. Binary by: Alan Rubright	HPMam	A program to manipulate settings and fonts on HP LaserJet printers and compatibles. Includes an Intuition interface and some sample picture files. Version 1.0, binary only, shareware. By: Steve Robb
Fred Fish Disk 135	TeXF	WheelChairSim	A wheelchair simulator developed as a project for the Technical Resource Centre and the Albert Children's Hospital, to allow the matching of a wheelchair joystick to a child's handicap and allow the child to practice using the chair in a safe (simulated) environment. Binary only. Author: Unknown, submitted by Dr. Mike Smith	Scrambler	A simple program that will encode/decode a text file into illegible gibberish, which resembles executable code, to evade prying eyes. Version 0.01, binary only. Author: Foster Hall	Synthmania	An interesting, very small (and very persistent!) musical piece. If you plan on stopping it without using three fingers, you better read the document file first! Binary only. By: Holger Lubitz
Fred Fish Disk 136	AsmToolBox	SmallC	SmallC is a rather small subset of the full C language. It is capable of compiling itself, and other small, useful programs. Requires an assembler and linker to complete the package and produce working executables. Source and binary. By: Ron Cain. Amiga port by Will Kusche.	Fred Fish Disk 149	AnimalSounds	Fred Fish Disk 154	Ada
AsmToolBox	Interfacing between assembler programs and AmigaDOS easy. With source. By: Warren Ring	SBProlog	Volume 1 of the 2 volume Stony Brook Prolog (SBP) distribution, V2.3.2. This volume contains the executables and libraries. Volume 2, on FF140, contains the C and Prolog Source. By: Logic Programming Group at SUNY, Stony Brook. Amiga port by David Roch & Scott Evenden	DX-VoiceSorter	Written to be used with Jack Deckard's VoiceFilter program. (Disk 82). It allows for the sorting of a number of voicefiles stored using that program into a new voicefile of voices made up from various files. Includes source. Author: David Bouckley	Ada	An Ada Syntax checker for the Amiga. Includes lex and yacc source. Author: Herman Fischer; updates by William Loftus
Bison	A replacement for unix 'yacc' command. From the GNU (GNU is Not Unix) effort. Port of the latest GNU version of bison's current features. Includes source & test pro. "calc". By: Bob Corbett and Rick Stallman	Fred Fish Disk 141	SBProlog	Like Unix "more", only better, with forward and backward scrolling, searching and positioning by percent of file and line number, etc. Now lets you also print the current file. Very useful! This is Amiga version 1.3, an update to the version on disk number 92. Includes source. Author: Mark Nudelmann, Amiga port by Bob Leivian	Keep	AssemblyDemos	An interesting group of assembly language demos for your visual and aural pleasure. Binary only. By: Foster Hall
HP2Pcs	Interactive puzzle prog. takes any IFF file containing up to 16 colors, and breaks it into squares to make a puzzle the user can then piece back together again. V1.1, update of FF122, includes source. By: Al Ozer	Fred Fish Disk 142	Diff	UnDelete	A nice little utility program with an intuition interface for BBS and network junkies who download messages in one large file and then read them off-line. Using only the mouse, you can drive through such files a message at a time, examine each at your leisure and tag those you wish to keep. Version 1.2, binary only, but source available with donation to author. Author: Tim Grantham	DiskLib	Two utilities for those people who like to split up PD disks into disks of different categories. Includes source. By: Wilson Snyder
Paste	Version of the Unix paste utility. Paste concatenates corresponding lines of the specified files into a single output line (horizontal or parallel merging) or concatenates them into alternate lines (vertical or serial merging). S. By: David Inhat	Fred Fish Disk 143	FracGen	WhereIs	Like Unix "more", only better, with forward and backward scrolling, searching and positioning by percent of file and line number, etc. Now lets you also print the current file. Very useful! This is Amiga version 1.3, an update to the version on disk number 92. Includes source. Author: Mark Nudelmann, Amiga port by Bob Leivian	Guardian	Another virus diagnosing and vaccination program. Recognizes any non-standard bootblock. Includes a small utility program to permanently place the program on a copy of your kickstart disk in place of the lemons (if ever) used Debug() function. Binary only. By: Leonardo Fei
YaBoing!	Game prog. demonstrating hardware sprite usage, including collision detection. Update of FF36. S. By: Al Ozer, based on original by Leo Schwab	Fred Fish Disk 144	SciSubr	PrintSpool	"Scheme is a statically scoped and properly tail-recursive dialect of the Lisp programming language invented by Guy Lewis Steele Jr. and Gerald Jay Sussman." Binary only. Amiga port by Ed Puckett	Utilities	A group of four little utility programs:
Zoo	File archiver, like "arc" in concept, but different in implementation and user interface details. Includes features that "arc" lacks (such as filepaths names up to 255 characters in length). V1.71, update of FF108. B. By: Rahul Dhesi, port by Brian Waters	Fred Fish Disk 145	Csh	AmigaBasic DC-10 instrument flight simulator. Appears to be quite in-depth with flight-planning and take-off options along with an extensive documentation file. Requires rebuilding on a separate disk and was successfully done so by following the author's instructions in the ReadMe_First file. Author: Jan Arkesteijn	DC10	UnDelete	Undelete a file from floppy (DFO) to any device you request, checks for a disk in the drive and allows you to abort cleanly with a CTRL 'C'.
Fred Fish Disk 137	CT	Fred Fish Disk 146	Blanker2	ExecLib	An update to the Airfoil generator on disk #71. Generates airfoil models as well as their corresponding streamline and pressure distributions. Includes source. Authors: Russell Leighton Addendum by David Foster	WhereIs	Looks for a file and/or directory defaults to the current device
CT	Program to display images from a CT scanner, along with several interesting sample images of scans of real people, including a skull, brain, heart, and spine. Each image is 256 by 256 pixels in 2048 gray scale. The display software, a primitive user interface, is quite powerful, including functions like convolutions, averaging, laplacians, unsharp masking, edge detection, gradients, etc. Binary. By: J. Harman	Fred Fish Disk 147	DMouse	Icons	AmigaBasic DC-10 instrument flight simulator. Appears to be quite in-depth with flight-planning and take-off options along with an extensive documentation file. Requires rebuilding on a separate disk and was successfully done so by following the author's instructions in the ReadMe_First file. Author: Jan Arkesteijn	CAL	- Clone of the Unix CAL, dates from year 1 to 9999.
JeanStones	Miscellaneous cute icons created for AMUC's monthly newsletter disk. Submitted by Stephen Vermeulen. Author: Steve Jeans	Fred Fish Disk 148	Net	PCOPY	A working example of how to build and use user-defined disk-resident libraries. Of special interest to developers working with Lattos C. Author: Alex Livshits	CDlock	Simple title bar clock/memory gauge with pop to front.
Muncho	A cute little program which plays a digitized sound sample from which you insert or remove a disk from your drive. If you don't like the sounds, you can replace them with your own. Binary only. By: Andrew Werth	Fred Fish Disk 149	Tab	SCT	A utility program that saves your current mouse pointer to a small icon. You can restore the pointer just by double-clicking on its icon. Allows for building a whole library of pointers and to use them whenever you want. Binary only. Author: Alex Livshits	VirusX	An update to the virus-detecting program of the same name on disk number 137. This version also checks for the Byte-Bandi strain. Version 1.21, includes source. By: Steve Tibbett
Sit	Update to the Set Icon Type prog. on FF107. V1.10, includes source. Author: Stephen Vermeulen	Fred Fish Disk 150	TinyProlog	SlideShow	A utility program that saves your current mouse pointer to a small icon. You can restore the pointer just by double-clicking on its icon. Allows for building a whole library of pointers and to use them whenever you want. Binary only. Author: Alex Livshits	VirusAlert!	Yet another anti-virus program with a twist. Once installed a message is displayed just after a warm or cold boot notifying the user that the disk and memory are virus-free, and forcing a mouse-button press before continuing. Anything writing to the bootblock thereafter will destroy the message and a normal virus-infected boot (???) will take place. Versions 1.01 and 2.01. Binary only. By: Foster Hall
VGad	A new gadget editor that takes two pictures of the window and its gadgets, one being the normal gadget state and the other being the fully selected state, then merges the data and converts to C source code. V1.0, binary only. Author: Stephen Vermeulen	Fred Fish Disk 151	Blanker2	UUCP	An implementation of the PILOT language for the Amiga, including a demo done for the National Park Service. PILOT is a limited use language for use in educational and computer based instruction programs. Binary only with Beta test kit available from authors. By: T. LaGrone	Wicon	A "Window iconifier". Allows you to turn your windows into small icons which can be later recalled. Currently installed with MacWin to give your windows a "rubber-banding" effect. Version 1.14, includes source. By: Steven Sweeting introducing the Amiga
VirusX	A boot sector virus check program that runs in the background and automatically checks all inserted disks for a nonstandard boot sector. Such disks can optionally have their boot sector rewritten to remove the virus. Includes source. Author: Steve Tibbett	Fred Fish Disk 152	Csh	RunBack	A small utility designed to be a direct replacement for NoFastMem kind of programs. It modifies the boot block of a disk, so when you boot with it, all memory allocations will return only CHIP memory. Author: Alex Livshits	WhereIs	Looks for a file and/or directory defaults to the current device
VLabel	Program to print fancy customized disk labels. Combines an IFF picture and up to 50 lines of text (which may be placed arbitrarily in any font or point size) then print the result. The IFF picture can be virtually any size (up to 1008 by 1000). It will also print labels from a batch file produced by SuperBase. V1.20, binary only. By: Stephen Vermeulen	Fred Fish Disk 153	DMouse	UUCP	A small utility designed to be a direct replacement for NoFastMem kind of programs. It modifies the boot block of a disk, so when you boot with it, all memory allocations will return only CHIP memory. Author: Alex Livshits	CAL	- Clone of the Unix CAL, dates from year 1 to 9999.
Fred Fish Disk 138	AmigaLine	Fred Fish Disk 154	Net	Fred Fish Disk 155	GlobeDemo	CDlock	Simple title bar clock/memory gauge with pop to front.
AmigaLine	A series of various technical notes for Amiga programmers. By: Bryce Nesbitt	Fred Fish Disk 155	Tab	GlobeDemo	Graphics demo displays very smooth transitions of the rotating earth. Pop-up menu. Source. By: Bob Cowin	VirusX	An update to the virus-detecting program of the same name on disk number 137. This version also checks for the Byte-Bandi strain. Version 1.21, includes source. By: Steve Tibbett
Diff	Uses the same algorithm as the Unix diff program and also produces context diffs, suitable for use with patch. Binary only. By: Unknown (Decus C diff?)	Fred Fish Disk 156	TinyProlog	Icons	Yet another potpourri of interesting icons to choose from if you need one for your own program. By: Dave Turnock	VirusAlert!	Yet another anti-virus program with a twist. Once installed a message is displayed just after a warm or cold boot notifying the user that the disk and memory are virus-free, and forcing a mouse-button press before continuing. Anything writing to the bootblock thereafter will destroy the message and a normal virus-infected boot (???) will take place. Versions 1.01 and 2.01. Binary only. By: Foster Hall
Foreach	A simple but useful program that expands a wild card file specification and then invokes the specified command once per expanded filename, with the expanded filename as the command argument. Includes source. Author: Jonas Flygare	Fred Fish Disk 157	Blanker2	PCOPY	A small intuition-based disk copier similar to the resident "DiskCopy" except with write-verify and other user-selectable options. Useful for making multiple copies with reliable data. Requires two disk drives. S. By: Dirk Reisig	NoSmoking	Sample program showing the use of a recoverable alert while displaying a personal health message. Includes source. Author: Theo Kernanidis
MacFont	A conversion tool to convert Mac fonts to Amiga fonts. Binary only. By: John O'Neill and Rico Mariani	Fred Fish Disk 158	DMouse	SCT	A CLI-based utility (SetColorTable) for displaying and/or setting a screen's colors. Save the colors of a screen to be restored later, or copy one screen's colors to another. Includes source. Author: aklevin	Scenery	A very nice assembly language random scenery generator. Generates very realistic looking landscapes. Includes intuition interface and lots of menu options. V. 1.0, binary only. By: Brett Casebolt
ModuTools	Various useful routines for those using in Modula on the Amiga. Update to FF94. S. By: Jerry Mack	Fred Fish Disk 159	Net	SlideShow	Very nicely done slide-show program written in assembly language. Features forward/backward presentation and creative screen wipes. Currently works only with IFF lo-res pictures. Executable only along with some new IFF pictures to have come my way. Shareware (\$16). Authors: Mike Kitztrick and Sheldon Templeton	Fred Fish Disk 156	Blocks2
vt100	Two new versions of Dave's vt100 terminal emulator. One version, based on vt100 2.6, has been enhanced by John Barsinger to include an iconify feature, add full 132 column support using overscan, and other features (binary only). The second version is release 2.8 of the main-stream version of vt100, as enhanced and supported by Tony Sumrall. S. By: Dave Wecker	Fred Fish Disk 160	Tab	UUCP	A CLI-based utility (SetColorTable) for displaying and/or setting a screen's colors. Save the colors of a screen to be restored later, or copy one screen's colors to another. Includes source. Author: aklevin	Flex	Amusing and colorful display of a moving trail of "blocks". Update to version on disk number 71, however this version also Source. By: Gary Walker
Fred Fish Disk 139	AmiCrn	Fred Fish Disk 161	TinyProlog	RunBack	A utility program that saves your current mouse pointer to a small icon. You can restore the pointer just by double-clicking on its icon. Allows for building a whole library of pointers and to use them whenever you want. Binary only. Author: Alex Livshits	Go64	Flex is a replacement for the Unix 'lex' (lexical analyzer generator) program that is faster than lex, and freer/distributable. Includes source. Authors: Jelf Poskanzer, Vern Paxson, et. al. Submissions by William Loftus and Scott Henry
AmiCrn	An enhanced and debugged version of AmiCrn 2.3 from FF113. Includes source. By: Steve Sampson, Rich Schaeffer, Christian Balzer	Fred Fish Disk 162	Blanker2	UUCP	A utility program that saves your current mouse pointer to a small icon. You can restore the pointer just by double-clicking on its icon. Allows for building a whole library of pointers and to use them whenever you want. Binary only. Author: Alex Livshits	Grammars	Another screen hack aimed at an earlier Commodore product (Not to be confused with the commercial product Go-64! from Software Insight Systems). Includes source. By: Joerg Anslk
ListScanner	A nice little utility to display all the Exec lists. Similar to Xplo utility FF73. Includes source in assembler. By: Helko Rath	Fred Fish Disk 163	C-Light	RunBack	A variant of Rob Peck's RunBackGround program from disk number 73. Allows you to start a new CLI program and run it in the background, then closes the new CLI. This version automatically searches the command-search-path to find the program. Source. By: Daniel Barrett	OOPS!	A group of lexical grammar files for Ada, C and Pascal for use in conjunction with the flex program on this disk and thebison program on disk #155. By: Various, submitted by William Loftus
ProCalc	Simulates HP-11C programmable calculator. Both English & German versions. Shareware. B only. By:	Fred Fish Disk 164	CrcLists	UUCP	A variant of Rob Peck's RunBackGround program from disk number 73. Allows you to start a new CLI program and run it in the background, then closes the new CLI. This version automatically searches the command-search-path to find the program. Source. By: Daniel Barrett	AmicForm	Tired of the monochrome background color of your Workbench or CL? Then try this colorful screen hack to brighten things up! Includes source. By: Joerg Anslk
		Fred Fish Disk 165	DmeMacros	RunBack	A little utility that opens a window on the current screen and displays information about the pointer. Allows for absolute or relative measurement between two points on the screen. Very handy for precise positioning of icons and such. Includes source. Author: Dirk Reisig	ArinBalls	Another screen hack aimed at an earlier Commodore product (Not to be confused with the commercial product Go-64! from Software Insight Systems). Includes source. By: Joerg Anslk
		Fred Fish Disk 166	MemoPad	UUCP	A little utility that opens a window on the current screen and displays information about the pointer. Allows for absolute or relative measurement between two points on the screen. Very handy for precise positioning of icons and such. Includes source. Author: Dirk Reisig		
		Fred Fish Disk 167	MicroGnuEmacs	RunBack	A version of uucp (Unix to Unix Copy) program for the Amiga, along with some miscellaneous support utilities like cron, mail, and compress. Includes source. Various, submitted by William Loftus		
		Fred Fish Disk 168	ModuTools	UUCP	A version of uucp (Unix to Unix Copy) program for the Amiga, along with some miscellaneous support utilities like cron, mail, and compress. Includes source. Various, submitted by William Loftus		
		Fred Fish Disk 169	vt100	RunBack	A variant of Rob Peck's RunBackGround program from disk number 73. Allows you to start a new CLI program and run it in the background, then closes the new CLI. This version automatically searches the command-search-path to find the program. Source. By: Daniel Barrett		
		Fred Fish Disk 170	AmiCrn	UUCP	A version of uucp (Unix to Unix Copy) program for the Amiga, along with some miscellaneous support utilities like cron, mail, and compress. Includes source. Various, submitted by William Loftus		
		Fred Fish Disk 171	ListScanner	RunBack	A variant of Rob Peck's RunBackGround program from disk number 73. Allows you to start a new CLI program and run it in the background, then closes the new CLI. This version automatically searches the command-search-path to find the program. Source. By: Daniel Barrett		
		Fred Fish Disk 172	ProCalc	UUCP	A version of uucp (Unix to Unix Copy) program for the Amiga, along with some miscellaneous support utilities like cron, mail, and compress. Includes source. Various, submitted by William Loftus		
		Fred Fish Disk 173		RunBack	A variant of Rob Peck's RunBackGround program from disk number 73. Allows you to start a new CLI program and run it in the background, then closes the new CLI. This version automatically searches the command-search-path to find the program. Source. By: Daniel Barrett		
		Fred Fish Disk 174		UUCP	A version of uucp (Unix to Unix Copy) program for the Amiga, along with some miscellaneous support utilities like cron, mail, and compress. Includes source. Various, submitted by William Loftus		
		Fred Fish Disk 175		RunBack	A variant of Rob Peck's RunBackGround program from disk number 73. Allows you to start a new CLI program and run it in the background, then closes the new CLI. This version automatically searches the command-search-path to find the program. Source. By: Daniel Barrett		
		Fred Fish Disk 176		UUCP	A version of uucp (Unix to Unix Copy) program for the Amiga, along with some miscellaneous support utilities like cron, mail, and compress. Includes source. Various, submitted by William Loftus		
		Fred Fish Disk 177		RunBack	A variant of Rob Peck's RunBackGround program from disk number 73. Allows you to start a new CLI program and run it in the background, then closes the new CLI. This version automatically searches the command-search-path to find the program. Source. By: Daniel Barrett		
		Fred Fish Disk 178		UUCP	A version of uucp (Unix to Unix Copy) program for the Amiga, along with some miscellaneous support utilities like cron, mail, and compress. Includes source. Various, submitted by William Loftus		
		Fred Fish Disk 179		RunBack	A variant of Rob Peck's RunBackGround program from disk number 73. Allows you to start a new CLI program and run it in the background, then closes the new CLI. This version automatically searches the command-search-path to find the program. Source. By: Daniel Barrett		
		Fred Fish Disk 180		UUCP	A version of uucp (Unix to Unix Copy) program for the Amiga, along with some miscellaneous support utilities like cron, mail, and compress. Includes source. Various, submitted by William Loftus		
		Fred Fish Disk 181		RunBack	A variant of Rob Peck's RunBackGround program from disk number 73. Allows you to start a new CLI program and run it in the background, then closes the new CLI. This version automatically searches the command-search-path to find the program. Source. By: Daniel Barrett		
		Fred Fish Disk 182		UUCP	A version of uucp (Unix to Unix Copy) program for the Amiga, along with some miscellaneous support utilities like cron, mail, and compress. Includes source. Various, submitted by William Loftus		
		Fred Fish Disk 183		RunBack	A variant of Rob Peck's RunBackGround program from disk number 73. Allows you to start a new CLI program and run it in the background, then closes the new CLI. This version automatically searches the command-search-path to find the program. Source. By: Daniel Barrett		
		Fred Fish Disk 184		UUCP	A version of uucp (Unix to Unix Copy) program for the Amiga, along with some miscellaneous support utilities like cron, mail, and compress. Includes source. Various, submitted by William Loftus		
		Fred Fish Disk 185		RunBack	A variant of Rob Peck's RunBackGround program from disk number 73. Allows you to start a new CLI program and run it in the background, then closes the new CLI. This version automatically searches the command-search-path to find the program. Source. By: Daniel Barrett		
		Fred Fish Disk 186		UUCP	A version of uucp (Unix to Unix Copy) program for the Amiga, along with some miscellaneous support utilities like cron, mail, and compress. Includes source. Various, submitted by William Loftus		
		Fred Fish Disk 187		RunBack	A variant of Rob Peck's RunBackGround program from disk number 73. Allows you to start a new CLI program and run it in the background, then closes the new CLI. This version automatically searches the command-search-path to find the program. Source. By: Daniel Barrett		
		Fred Fish Disk 188		UUCP	A version of uucp (Unix to Unix Copy) program for the Amiga, along with some miscellaneous support utilities like cron, mail, and compress. Includes source. Various, submitted by William Loftus		
		Fred Fish Disk 189		RunBack	A variant of Rob Peck's RunBackGround program from disk number 73. Allows you to start a new CLI program and run it in the background, then closes the new CLI. This version automatically searches the command-search-path to find the program. Source. By: Daniel Barrett		
		Fred Fish Disk 190		UUCP	A version of uucp (Unix to Unix Copy) program for the Amiga, along with some miscellaneous support utilities like cron, mail, and compress. Includes source. Various, submitted by William Loftus		
		Fred Fish Disk 191		RunBack	A variant of Rob Peck's RunBackGround program from disk number 73. Allows you to start a new CLI program and run it in the background, then closes the new CLI. This version automatically searches the command-search-path to find the program. Source. By: Daniel Barrett		
		Fred Fish Disk 192		UUCP	A version of uucp (Unix to Unix Copy) program for the Amiga, along with some miscellaneous support utilities like cron, mail, and compress. Includes source. Various, submitted by William Loftus		
		Fred Fish Disk 193		RunBack	A variant of Rob Peck's RunBackGround program from disk number 73. Allows you to start a new CLI program and run it in the background, then closes the new CLI. This version automatically searches the command-search-path to find the program. Source. By: Daniel Barrett		
		Fred Fish Disk 194		UUCP	A version of uucp (Unix to Unix Copy) program for the Amiga, along with some miscellaneous support utilities like cron, mail, and compress. Includes source. Various, submitted by William Loftus		
		Fred Fish Disk 195		RunBack	A variant of Rob Peck's RunBackGround program from disk number 73. Allows you to start a new CLI program and run it in the background, then closes the new CLI. This version automatically searches the command-search-path to find the program. Source. By: Daniel Barrett		
		Fred Fish Disk 196		UUCP	A version of uucp (Unix to Unix Copy) program for the Amiga, along with some miscellaneous support utilities like cron, mail, and compress. Includes source. Various, submitted by William Loftus		
		Fred Fish Disk 197		RunBack	A variant of Rob Peck's RunBackGround program from disk number 73. Allows you to start a new CLI program and run it in the background, then closes the new CLI. This version automatically searches the command-search-path to find the program. Source. By: Daniel Barrett		
		Fred Fish Disk 198		UUCP	A version of uucp (Unix to Unix Copy) program for the Amiga, along with some miscellaneous support utilities like cron, mail, and compress. Includes source. Various, submitted by William Loftus		
		Fred Fish Disk 199		RunBack	A variant of Rob Peck's RunBackGround program from disk number 73. Allows you to start a new CLI program and run it in the background, then closes the new CLI. This version automatically searches the command-search-path to find the program. Source. By: Daniel Barrett		
		Fred Fish Disk 200		UUCP	A version of uucp (Unix to Unix Copy) program for the Amiga, along with some miscellaneous support utilities like cron, mail, and compress. Includes source. Various, submitted by William Loftus		



BootBack	A handy little utility to copy and save the boot block from a disk, then later restore it should the disk get stomped on by some ugly virus. Source: by: David Joiner	Nag	A shareware appointment calendar with it's own editor and a unique 'nagging' feature utilizing the Amiga's voice and audio devices. V.1.6, binary only, by: Richard Lee Stockton	names up to 255 characters in length). This is version 2.00, an update to FF 136. Binary only. by: Raul Dhesi, Amiga port by Brian Waters	Fred Fish Disk 170	Aftermath	Communications program utilizing IBM 3278 terminal emulation. Binary only. by: Don Brereton			
ECPM	A CPM emulator for the Amiga. Emulates an 8080 along with H19 terminal emulation. Update from version on disk number 109. Source: by: Jim Cathey; port by Charlie Gibbs; Significant improvements by Will Kusche	Perl	Practical Extraction and Report Language, an interpreted language optimized for scanning arbitrary text files, extracting information from those text files, and printing reports based on the information. by: Larry Wall	Fred Fish Disk 165	Comman	Extremely useful replacement for the standard console handler, provides line editing and command line histories. Completely transparent to any application program that uses CON: windows. Shareware, and well worth a donation to the author. V.1.3, binary only, update to FF133, by: William Hawes	Dis6502	A ported 6502 assembler with support added for C64 binary files. Includes source. by: Robert Bond, Amiga port by Ulf Finkelnstein		
KeyFiler	BBS message file sorter that allows sorting by keyword. Includes a textreader, Soundex matching, and limited wildcard capabilities. V.1.0, Binary only: by: J. Mottisinger	VRTest	Another anti-virus utility that allows visual inspection of ram starting a \$7F7E, ram clearing, bootblock inspection and vector monitoring/rewriting. Written in assembly V.3.2, binary only: by: Babar Khan	CPM	Another CPM emulator independently authored from the version that appeared on disk #157. Emulates a CPM computer with a Z80 processor connected to an ADM3A terminal. Assembly source included. by: Ulf Nordquist	FastText	Blitter based fast text rendering routines written in assembly. Unique in the fact that they speed up rendering of non-proportional fonts of any height, and from 4-16 pixels in width. Source and test program included. by: Darren M. Greenwald			
ScreenZap	A little utility to clean away screens that are left by illbehaving programs. It will kill every screen behind the WorkBench, noting how many it gets. The screens in front of WB are not affected. Source: by: Lars Clausen	XBoot	Very simple utility to convert a boot block into an executable file to use your favorite debugger (Wack, Dis, etc.) to study it. withsource: by: Francois Rouaix	Parnag	A program to aid in performing color separations on Epson JX-80 printers. Source: by: John Hodgson	MRBackUp	A hard disk backup utility that does a file by file copy to standard AmigaDOS floppy disks. Includes an intuition interface and file compression. Version 2.4, Update to FF129. Binary only: by: Mark Rinfret			
SePrefs	Allows you to build a whole library of preference settings and instantly switch back and forth between them. Affects all preference settings not just the colors. Very useful for machines with multiple users or multiple external devices. Includes Amiga's default and various sample preference settings. B only: by: Martin Hippel	Fred Fish Disk 162	Avi	A workalike version of the UNIX vi editor for the Amiga. Though not especially recommended for beginners, designed for those of you who may have the vi commands permanently hard-coded into your fingertips! V.1.0, binary only: by: Peter Nestor	PlotView	A couple of programs, PlotView and Plot2Am, for viewing UNIX plot files. Also included are two sub-directories: Plot — a device independent plotting package for the Amiga, compatible with the UNIX plot subroutine package and Plot2Tek — converts UNIX plot format files to Tektronix 410x terminal graphic commands. Source included. Author: Joel Swank	PrAnim	Nifty pointer animation program, includes lots of samples, a utility program and instructions on creating your own animations. Binary only, shareware: by: Tim Kemp		
Xicon	Xicon lets you use icons to call up scripts containing CLI commands. This is version 2.01, an update to FF102. Includes source: by: Pete Goodeve	CLI Utilities	This directory contains several subdirectories with small utilities, collected from various sources, only usable from the CLI. Some with source. Author: Various	RamCopy	A copy program designed for machines with 1 meg or more of Ram and only one disk drive. Copy a complete disk in only one pass. by: Stephen Gunn	Surf	Generates better surfaces of revolution. Will produce some amazing pictures of wineglasses, doorknobs, or other objects one could turn on a lathe. Includes the capacity to map IFF image files onto any surface that it can draw. Source included: by: Eric Davies			
Fred Fish Disk 158	DisX	Nicely done Sector-based disk editor. Binary only: by: Steve Tibbett	Dark	A small graphics and animation demo withsource: by: Phil Robertson	SPUDClock	A simple program that uses the narrator device to speak the time at certain user specified intervals. Lots of command line options. Version 1.2, includes source. Authors: Robert E. Beatty and H. Bret Young	Turbo	Opens a small window with a gadget that when selected, turns off bilplane, sprite, copper and audio DMA, to increase system speed. Source: by: Oliver Wagner		
MemBoardTest	Originally designed for production testing of A1000 memory boards. Very nice intuition interface. Version 2.4, Source in Modula by: George Vokalek	Flow2Trot	A little utility to convert from New Horizons Software "FLOW" files to UNIX "trif" files, suitable for printing on any troff-compatible laser printer. V. 1.0, includes source and a sample "FLOW" file. by: Daniel Barrett	Fred Fish Disk 166	AutoGraf	Collects and graphically displays information on auto miles. Features such as miles per gallon, cost per mile, miles driven, high, low, averages, etc. Includes sample data file, a couple utility programs and source. Version 1.0 Author: Joel Swank	Fred Fish Disk 171	AZComm	Modified version of Comm 1.34 that contains Zmodem send, receive, and resume receive. Version 1.00, Binary only: by: SS. Pale, based on Comm 1.34 by DJ James	
MSDOS	A program to list files written in standard MS-Dos or Atari ST format. The files can then be copied to Ram and rewritten to disk in Amiga-Dos format. Binary only, Shareware, V.0.1. Author: Frank Wibelberg	Labyrinth	A shareware role-playing text adventure game similar in operation to the Infocom text adventures. Includes source: by: Russell Wallace	Crel	Cross reference program. Prints out your code with line-numbers and complete key-word cross-referencing. Update to #ff103 which had a serious bug. Includes source: by: Mike Edmonds; Amiga port by Joel Swank	Maze	A couple of very nice demos for the creation and use of single-source mazes, one of which is practically a stand-alone game. Includes source: by: Werner Gunther			
PCBTool	Early version of a shareware PC board layout program. Lots of options including variable size pads and traces, grids, grid snap, layers, zoom, selectable centering, text and more. This version does not support printer/plotter dumps or libraries. V.2.6, B only: by: George Vokalek	litar	Maintains archives of Interchange File Format (IFF) FORM CAT and LIST files in a manner that complies with the IFF CAT specification. V.1.2, includes source by: Karl Lehenbauer	MultCalc	Yet another RPN type graphic calculator. This one generates answers with extreme precision (if 3000 digits is enough). Features a 48-digit scrollable display, mouse driven with lots of keyboard shortcuts, and iconification during non-use. Binary: by: Ken Johnson	Scobon-C	Atari ST version of what appears to be a full K&R freeware C-compiler, assembler and linker. The compiler main pass and the assembler were compiled and tested on an Amiga A2000 with only minimal changes, and they appear to work (to the extent that they believe they are running on an Atari-ST), so an Amiga port should be relatively easy. by: Scobon, Limited.			
ScreenX	A handy little background utility that provides a small clock/memory counter in its inactive mode and a versatile screen manipulator when called upon. Binary, source available from author. V2.1. by: Steve Tibbett	SetPALonNTSC	A couple of utility programs for testing the suitability of a developed program in either the PAL or NTSC environments. Includes source and a sample program. by: Peter Kite	Stavie	A public domain clone of the UNIX 'vi' editor. Supports window-zooming, arrow keys, and the help key. Version 3.10a, includes source. Amiga port by Tony Andrews	Xoper	Very comprehensive program to monitor and control system activity. Monitor cpu, memory usage, ports, interrupts, devices. Closes windows, screens, show loaded fonts or last guru code numbers. Clean up memory, flush unused libraries, devices, fonts, etc. and a whole bunch more! Spawns its own process. A very handy background task to have loaded. Assembly source included: by: Werner Gunther			
TaskX	A "real-time" task editor. Lets you list and set priorities of all currently running tasks. Binary: V. 2.0, by: S. Tibbett	TES	"The Electronic Slave" adds a gadget strip to the top of the cli window to perform such functions as device directories into, run ED, and time. Currently, assignments are hard-coded but not difficult to change if you own a compiler. V. 1.1, with source: by: Joerg Anisk	Fred Fish Disk 167	CDed	English to C (and vice versa) translator for C declarations. This little gem will translate english such as "declare foo as pointer to function returning pointer to array 10 of pointer to long" into "long foo[10];" and vice versa. Update to FF114. Includes source: by: Graham Ross with enhancements by David Wolverson, Tony Hansen, Merlyn LeRoy, Ulf Finkelnstein & more.	Handshake	A full featured VT52/VT100/VT102/VT220 terminal emulator. The author has taken great pains to support the full VT102 spec. Now supports ANSI codes, screen capture and more. Update to version on FF60. Version 2.12a, binary only, shareware: by: Eric Haberlertner		
VirusX	Update to FF154, checks for a couple of additional new strains. Includes source V. 1.6 by: Steve Tibbett	UnknownGirl	Small musical piece similar in execution to Synthesia on FF153. Binary only. by: Holger Lubitz	CLcon	Run CLI programs from workbench, similar in operation to IconExec, but more versatile. by: Bryan Ford	CloseMe	Another ingenious perversion in the screen hack category. Don't miss this one...surely destined to become a classic! Includes source: by: Charlie Gibbs (Dynamic Sound Machine) Demo version with "Save" disabled of a program which will take any IFF sound or raw data and save it as a totally self-contained, runnable program. by: Foster Hall	Mfx	Small program to insert in the startup-sequence of the commercial program, Marauder II, from Discovery Software, International. When the copy process starts, the rainbow screen is covered by a bare screen until the copy is finished. Author claims a 25% decrease in copy time is achieved. V.1.0, Binary: by: Stephen Gunn	
YachtC3	Update to FF110, contains some fixes and incorporates a simple sound process. V3, includes source. Author: Sheldon Leemon, with enhancements by Mark Schretlen	Bankin	A complete shareware checkbook system. Update to FF120. V1.5, binary: by: Hal Carter	DSM	Dynamic Sound Machine Demo version with "Save" disabled of a program which will take any IFF sound or raw data and save it as a totally self-contained, runnable program. by: Foster Hall	MRPrint	A oil-based text file printing utility with lots of nice features, including tab-to-space expansion, page headers, line numbers, margin control with line-splitting and pagination control, ARP wildcard support, and auto-rejection of files containing binary characters. Version 3.1, includes source. Author: Mark Rinfret	ProCalc	Simulates an HP-11C programmable calculator. Lots of enhancements and bug fixes since the original version on FF139. V.1.2, binary only, shareware: by: Goz Muller	
Fred Fish Disk 159	Free	A little command to put in your c directory that returns memory status and number of tasks currently served by EXEC. Includes source: by: Joerg Anisk	FiveLine	Fast-paced & addictive! Source: by: Neil Fiskekin	SMus3.6a	An enhanced version of the smus player that last appeared on FF58. Author: John Hodgson	Spiff	Make controlled approximations between two files. Similar to "diff" but more versatile. Allows for the handling of numerals as string literals or numeric values with adjustable tolerances. Provides for embedded commands, scriptfiles, and many other command-line parameters. Potentially very useful, needs some Amiga-specific work. Source and example files included: by: Dan Nachbar, Bell Communications Research (BELLCORE)		
MidTools	A group of several different utility programs for those who run a Mid system. Binary only: by: Jack Deckard	Machin	A "mouse accelerator" program which includes hotkeys, the features of sun mouse, clicktofront, popit, title bar clock with a bbs online charge accumulator, and more. Update to FF130. V2.4c, binary only: by: Brian Motts	View	A mouse-oriented text file reader. Sample operation is demonstrated in reading the View.doc file, instead of using the usual "Less" text file reader. by: Bryan Ford	Fred Fish Disk 168	Fred Fish 168 and Fred Fish 169 contain programs submitted by Matt Dillon must be purchased at the same time to utilize the programs on the disk. In order to maintain the congruity of Matt Dillon's files, Fred kept the disks in their original form. We apologize for this inconvenience, but we feel the library should remain intact under Fred's original guide lines.	To Be Continued.....		
StarChart	Nicely done intuition based program to display and identify about 600 stars, galaxies and nebulae visible in the Northern hemisphere. V.1.2, Source: by: Ray R. Larson	MemTrace	Routines help debug memory allocation and freeing during program development. Complaints if you try to free memory you didn't allocate & reports on memory not freed when your prog finishes. by: Jojo Wesener	PCPatch	Patches for PCCopy and PCFormat from the "EXTRAS 1.2" disk, to allow reading/writing/formatting 3.5 inch 360K (2 sides/40 Tracks/9 sectors) MS-DOS disks. by: Werner Gunther	Sounddemos	Some very nice demos for showing off the incredible audio power of the Amiga! 100% assembly language, Connect the stereo for these! Author: Foster Hall	In Conclusion	To the best of our knowledge, the materials in this library are freely distributable. These materials were either publicly posted and placed in the public domain by their authors, or they have restrictions published in their files to which we have adhered. If you become aware of any violation of the authors' wishes, please contact us by mail.	
TaskControl	Nicely done task-handling program allowing you to put to sleep, kill or change priorities of the all the currently loaded tasks. Also potentially GURU-producing, so be careful what tasks you kill, change priorities of, etc. Handy windowkiller will reduce it almost to an icon. Binary only: by: J. Martin Hippel	PoPatch	Patches for PCCopy and PCFormat from the "EXTRAS 1.2" disk, to allow reading/writing/formatting 3.5 inch 360K (2 sides/40 Tracks/9 sectors) MS-DOS disks. by: Werner Gunther	ReadmeMaster	A nifty little database for finding those programs that you know exist somewhere (???) in the AmigaLib disk library. Maintains a key-word dictionary of the Contents descriptions that allows searching by disk number, program title, author's name, or some other descriptive word. Currently supports disks 1-154 with planned updates. Binary: by: Harold Morash	DISM V1.10	Mouse enhancer - acceleration, blanking, etc...now handles requests better w/ auto-activate by M Dillon	IMPORTANT NOTICE!	This list is compiled and published as a service to the Commodore Amiga community for informational purposes only. Its use is restricted to non-commercial groups only! Any duplication for commercial purposes is strictly forbidden. As a part of Amazing Computing™, this list is inherently copyrighted. Any infringement on this proprietary copyright without expressed written permission of the publishers will incur the full force of legal actions.	
TUC	"The Ultimate Clock". Another window title clock/memory minder. This one is in 132 columns! Also gives the free memory on drives DF0, DF1 & DF2. Includes source. by: Joerg Anisk	ReadmeMaster	A nifty little database for finding those programs that you know exist somewhere (???) in the AmigaLib disk library. Maintains a key-word dictionary of the Contents descriptions that allows searching by disk number, program title, author's name, or some other descriptive word. Currently supports disks 1-154 with planned updates. Binary: by: Harold Morash	View	A mouse-oriented text file reader. Sample operation is demonstrated in reading the View.doc file, instead of using the usual "Less" text file reader. by: Bryan Ford	Fred Fish Disk 169	Fred Fish 168 and Fred Fish 169 contain programs submitted by Matt Dillon must be purchased at the same time to utilize the programs on the disk. In order to maintain the congruity of Matt Dillon's files, Fred kept the disks in their original form. We apologize for this inconvenience, but we feel the library should remain intact under Fred's original guide lines.	Any non-commercial Amiga user group wishing to duplicate this list should contact:		
Fred Fish Disk 160	Calis	A little utility to help analyze the flow of a C-program by laying out the functions called in a hierarchical manner. Originally from Usenet with major revisions by Kevin Braunsdorf, Amiga port by George MacDonald	C-Functions	A group of four little C-functions to add to your library to make your programming life a little easier. Includes source and a small demo program showing some of the results. by: Lars Thuring	DISM V1.10	Mouse enhancer - acceleration, blanking, etc...now handles requests better w/ auto-activate by M Dillon	SUPILIB	Support (link time) library required to compile Matt Dillon's programs. by M Dillon	PIM Publications, Inc.	
Check	A useful little utility for finding structural errors in C-source code. Many command-line options. V.1.03, binary only: by: Keith Elbertson	DiskSalv	Very useful program to recover files from a trashed AmigaDOS disk. Can also "undelete" files deleted by mistake, so long as they have not been overwritten by further disk activity. Requires two disk drives. Many enhancements since the original version on disk #20. Version 1.3, Binary only: by: Dave Haynie	Backup V2.01	HD backup/restore by M Dillon	LIBREF	Utility for generating run-time library link library assembly files, i.e., aasm, & the run-time library's vector list. Currently very Aztec in what it generates. by MDillon	DRES V1	Small-systems assembler. Handles 6502 and some of the Motorola single chip microcomputers. Generates code (not object module oriented). by Matt Dillon	P.O. Box 869
Dis	A 68000 disassembler, written in assembly, this is an update to the version on disk #128. Includes source: by: Greg Lee with enhancements by Will Kusche	Hed	A handy little editor that is more user-friendly than "Ed", yet doesn't require the memorization of complicated keystrokes of some of the larger, more powerful, editors. Binary only: by: Hal Carter	SUPILIB	Support (link time) library required to compile Matt Dillon's programs. by M Dillon	DASM V2.11	Small-systems assembler. Handles 6502 and some of the Motorola single chip microcomputers. Generates code (not object module oriented). by Matt Dillon	FILES V1.2	Matt Dillon's disk catalog program. by MDillon	Fall River, MA 02722
DMouse	A versatile screen & mouse blaster, auto window activator, mouse accelerator, popit, pop window to front, push window to back, etc, widget. V. 1.09, includes source. Update to FF 145 by: Matt Dillon	Newton	Uses the "Newton's Method" algorithm to estimate both real and imaginary roots of a polynomial of degree 20 or less. Version 1.0, includes source: by: Daniel Barrett	Backup V2.01	HD backup/restore by M Dillon	FTOHEX	Part of DASM used to convert a DASM executable into an intel-hex formatted ascii file. by Matt Dillon	SHELL V2.10	Added environment variable support and more. Use ConMan if you want command line editing by MDillon	PIM Publications, Inc.
DWIP	"Daisy Wheel IFF Printer". A graphics printing utility that allows the printing of IFF pictures on a daisy wheel printer. Includes source: by: Ken Van Camp	NewZAP	A third-generation multi-purpose file sector editing utility, from the author of FileZAP. Displays and edits full 512-byte sectors via a 106 character wide internal font. Includes a search feature to find specific strings or hex digits, forwards or backwards. Update to FF58. Version 3.18, Binary only: by: John Hodgson	FILES V1.2	Matt Dillon's disk catalog program. by MDillon	FINDIT V1.00	Search for a filename (wildcard) by MDillon	ASCM V1.00	Utility to add CR's before LF's in files by MDillon	is extremely interested in helping any Amiga user groups in non-commercial support for the Amiga.
M4	A UNIX M4 look-alike macro processor intended as a front end Ratfor, Pascal, and other languages that do not have a built-in macro processing capability. Pd M4 reads standard input, the processed text is written on the standard output. by: Ozan S. Yigit (oz)	PcView	Provides the PC community with the opportunity to display IFF pictures to the best of EGA's ability. Displays Amiga pics, IBM-PC Deluxe Paint PICS, Apple II-GS Deluxe Paint PICS, and others in the IFF standard format. Includes source. Author: John Hodgson	REMCOR V1.00	Utility to remove CR's in files by MDillon	LIBS V1.00	List libraries/ides in ram or attempt to remove libraries. by MDillon	SCAT V1.00	Utility to 'cat' binaries without blowing up the display by MDillon	
MemoPad	A shareware intuition-based memo reminder program. Nicely done. Update to version on disk #146, v. 2, binary only: by: Michael Griebeling	PrntDrivers	A couple of new Printer Drivers. Digital Equipment's LN03+ laser printer, Mannesmann Tally's MT420d dot matrix. Authors: DEC LN03 — Bernie Mentink MT420d — Sascha Wildner	CMP V1.00	Utility to compare two files by MDillon	ADOCR V1.00	Utility to add CR's before LF's in files by MDillon			
NeuralNets	A neural network example using the general-ized back-propagation delta rule for learning, specifically applied to the tabularata Little Red Riding Hood instance. by: J. C. Hoskins	Zoo	A file archiver, much like "arc" in concept, but different in implementation and user interface details. Includes some nice features that "arc" lacks (such as file path							



# Amazing Computing Reader Service Card

Want to know more about Amazing computing advertisers and their products? Here's your chance! Fill out the card below (for instructions, see page 96) and drop it in the mail (U.S. only). We will process your request promptly and forward it to the advertiser.

Quick easy, and efficient—just like all services of Amazing Computing: Your Original Amiga Monthly Resource.

## Amazing Computing Reader Service Card

Name \_\_\_\_\_  
Street \_\_\_\_\_  
City \_\_\_\_\_ St. \_\_\_\_\_ Zip \_\_\_\_\_  
Telephone \_\_\_\_\_

February card valid until 4/30/89  
see page 96 for references

### A. Do you own an Amiga?

- ☐ 1. Amiga 1000  
☐ 2. Amiga 500  
☐ 3. Amiga 2000

- ☐ 4. Soon  
☐ 5. Not Yet  
☐ 6. Just Looking

### B. What Amiga hardware product do you plan to buy next?

- ☐ 1. Amiga 500  
☐ 2. Amiga 2000  
☐ 3. Memory Expansion  
☐ 4. Hard Drive  
☐ 5. IBM Emulators  
☐ 6. Printer  
☐ 7. Modem  
☐ 8. Music Tool  
☐ 9. Video Product  
☐ 10. Other

### C. What Amiga software product do you plan to buy next?

- ☐ 1. C Language  
☐ 2. Fort Language  
☐ 3. Module-2 Language  
☐ 4. Assembly Language  
☐ 5. BASIC Language  
☐ 6. Entertainment  
☐ 7. Telecommunications  
☐ 8. Spreadsheet  
☐ 9. Database  
☐ 10. Financial  
☐ 11. Video  
☐ 12. Graphics  
☐ 13. Music  
☐ 14. Other

### D. Where do you buy your Amiga products?

- ☐ 1. Local Amiga Dealer  
☐ 2. Discount Department Store  
☐ 3. Manufacturer  
☐ 4. Mail Order

### E. Which type of articles do you like to see in Amazing Computing?

- ☐ 1. C Language  
☐ 2. Fort Language  
☐ 3. Module-2 Language  
☐ 4. Assembly Language  
☐ 5. BASIC Language  
☐ 6. Game Reviews  
☐ 7. Business Reviews  
☐ 8. Hardware Product Reviews  
☐ 9. Software Product Reviews  
☐ 10. Programming How To's  
☐ 11. Video Articles  
☐ 12. Graphics Articles  
☐ 13. Music Articles  
☐ 14. Hardware How To's  
☐ 15. PDS Updates  
☐ 16. Interviews  
☐ 17. Other

### F. Which articles would you like to see more of in Amazing Computing?

- ☐ 1. C Language  
☐ 2. Fort Language  
☐ 3. Module-2 Language  
☐ 4. Assembly Language  
☐ 5. BASIC Language  
☐ 6. Game Reviews  
☐ 7. Business Reviews  
☐ 8. Hardware Product Reviews  
☐ 9. Software Product Reviews  
☐ 10. Programming How To's  
☐ 11. Video Articles  
☐ 12. Graphics Articles  
☐ 13. Music Articles  
☐ 14. Hardware How To's  
☐ 15. PDS Updates  
☐ 16. Interviews  
☐ 17. Other

### G. Are you a subscriber to Amazing Computing?

- ☐ 1. Yes  
☐ 2. No

101

101	102	103	104	105	201	202	203	204	205
106	107	108	109	110	206	207	208	209	210
111	112	113	114	115	211	212	213	214	215
116	117	118	119	120	216	217	218	219	220
121	122	123	124	125	221	222	223	224	225
126	127	128	129	130	226	227	228	229	230
131	132	133	134	135	231	232	233	234	235
136	137	138	139	140	236	237	238	239	240
141	142	143	144	145	241	242	243	244	245
146	147	148	149	150	246	247	248	249	250
151	152	153	154	155	251	252	253	254	255
156	157	158	159	160	256	257	258	259	260
161	162	163	164	165	261	262	263	264	265
166	167	168	169	170	266	267	268	269	270
171	172	173	174	175	271	272	273	274	275
176	177	178	179	180	276	277	278	279	280
181	182	183	184	185	281	282	283	284	285
186	187	188	189	190	286	287	288	289	290
191	192	193	194	195	291	292	293	294	295
196	197	198	199	200	296	297	298	299	300

## Amazing Computing Reader Service Card

Name \_\_\_\_\_  
Street \_\_\_\_\_  
City \_\_\_\_\_ St. \_\_\_\_\_ Zip \_\_\_\_\_  
Telephone \_\_\_\_\_

February card valid until 4/30/89  
see page 96 for references

### A. Do you own an Amiga?

- ☐ 1. Amiga 1000  
☐ 2. Amiga 500  
☐ 3. Amiga 2000

- ☐ 4. Soon  
☐ 5. Not Yet  
☐ 6. Just Looking

### B. What Amiga hardware product do you plan to buy next?

- ☐ 1. Amiga 500  
☐ 2. Amiga 2000  
☐ 3. Memory Expansion  
☐ 4. Hard Drive  
☐ 5. IBM Emulators  
☐ 6. Printer  
☐ 7. Modem  
☐ 8. Music Tool  
☐ 9. Video Product  
☐ 10. Other

### C. What Amiga software product do you plan to buy next?

- ☐ 1. C Language  
☐ 2. Fort Language  
☐ 3. Module-2 Language  
☐ 4. Assembly Language  
☐ 5. BASIC Language  
☐ 6. Entertainment  
☐ 7. Telecommunications  
☐ 8. Spreadsheet  
☐ 9. Database  
☐ 10. Financial  
☐ 11. Video  
☐ 12. Graphics  
☐ 13. Music  
☐ 14. Other

### D. Where do you buy your Amiga products?

- ☐ 1. Local Amiga Dealer  
☐ 2. Discount Department Store  
☐ 3. Manufacturer  
☐ 4. Mail Order

### E. Which type of articles do you like to see in Amazing Computing?

- ☐ 1. C Language  
☐ 2. Fort Language  
☐ 3. Module-2 Language  
☐ 4. Assembly Language  
☐ 5. BASIC Language  
☐ 6. Game Reviews  
☐ 7. Business Reviews  
☐ 8. Hardware Product Reviews  
☐ 9. Software Product Reviews  
☐ 10. Programming How To's  
☐ 11. Video Articles  
☐ 12. Graphics Articles  
☐ 13. Music Articles  
☐ 14. Hardware How To's  
☐ 15. PDS Updates  
☐ 16. Interviews  
☐ 17. Other

### F. Which articles would you like to see more of in Amazing Computing?

- ☐ 1. C Language  
☐ 2. Fort Language  
☐ 3. Module-2 Language  
☐ 4. Assembly Language  
☐ 5. BASIC Language  
☐ 6. Game Reviews  
☐ 7. Business Reviews  
☐ 8. Hardware Product Reviews  
☐ 9. Software Product Reviews  
☐ 10. Programming How To's  
☐ 11. Video Articles  
☐ 12. Graphics Articles  
☐ 13. Music Articles  
☐ 14. Hardware How To's  
☐ 15. PDS Updates  
☐ 16. Interviews  
☐ 17. Other

### G. Are you a subscriber to Amazing Computing?

- ☐ 1. Yes  
☐ 2. No

102

101	102	103	104	105	201	202	203	204	205
106	107	108	109	110	206	207	208	209	210
111	112	113	114	115	211	212	213	214	215
116	117	118	119	120	216	217	218	219	220
121	122	123	124	125	221	222	223	224	225
126	127	128	129	130	226	227	228	229	230
131	132	133	134	135	231	232	233	234	235
136	137	138	139	140	236	237	238	239	240
141	142	143	144	145	241	242	243	244	245
146	147	148	149	150	246	247	248	249	250
151	152	153	154	155	251	252	253	254	255
156	157	158	159	160	256	257	258	259	260
161	162	163	164	165	261	262	263	264	265
166	167	168	169	170	266	267	268	269	270
171	172	173	174	175	271	272	273	274	275
176	177	178	179	180	276	277	278	279	280
181	182	183	184	185	281	282	283	284	285
186	187	188	189	190	286	287	288	289	290
191	192	193	194	195	291	292	293	294	295
196	197	198	199	200	296	297	298	299	300





NO POSTAGE  
NECESSARY  
IF MAILED  
IN THE  
UNITED STATES

**BUSINESS REPLY MAIL**

FIRST CLASS PERMIT NO. 36 FALL RIVER, MA

POSTAGE WILL BE PAID BY ADDRESSEE

*Amazing*  
COMPUTING

P.O. Box 869

Fall River, MA 02722-0869



NO POSTAGE  
NECESSARY  
IF MAILED  
IN THE  
UNITED STATES

**BUSINESS REPLY MAIL**

FIRST CLASS PERMIT NO. 36 FALL RIVER, MA

POSTAGE WILL BE PAID BY ADDRESSEE

*Amazing*  
COMPUTING

P.O. Box 869

Fall River, MA 02722-0869





# Do It NOW!!!

Subscribe TODAY and save over 49% on Amazing Computing!

*Amazing Computing is available to United States subscribers at \$24.00 for 12 issues. That's a savings of \$23.40 off the newsstand price. Canada and*

*Mexico, \$36.00. Foreign Surface to all other countries, \$44.00*

*(Air mail and first class rates available upon request.)*

*Enter my 12 issue subscription today!*

Name \_\_\_\_\_

Street \_\_\_\_\_

City \_\_\_\_\_ St. \_\_\_\_\_ Zip \_\_\_\_\_

Amount Enclosed \_\_\_\_\_

Sorry, we cannot accept credit cards or requests for billing at this time.  
All funds must be in U.S. Currency drawn on a U.S. Bank.

Please allow four to six weeks for processing.

# Save Time and Money!

Subscribe TODAY and save over 49% on Amazing Computing!

*Amazing Computing is available to United States subscribers at \$24.00 for 12 issues. That's a savings of \$23.40 off the newsstand price. Canada and*

*Mexico, \$36.00. Foreign Surface to all other countries, \$44.00.*

*(Air mail and first class rates available upon request.)*

*Enter my 12 issue subscription today!*

Name \_\_\_\_\_

Street \_\_\_\_\_

City \_\_\_\_\_ St. \_\_\_\_\_ Zip \_\_\_\_\_

Amount Enclosed \_\_\_\_\_

Sorry, we cannot accept credit cards or requests for billing at this time.  
All funds must be in U.S. Currency drawn on a U.S. Bank.

Please allow four to six weeks for processing.

## Clip and \$AVE !!

(over 49% of the newsstand price!)



**Please Mail to:**

**Amazing**  
COMPUTING

P.O. Box 869

Fall River, MA 02722-0869

Place this order form in an envelope with your  
Check or Money Order.

**Clip and \$AVE !!**

*(over 49% of the newsstand price!)*

**Please Mail to:**

**Amazing**  
COMPUTING

P.O. Box 869

Fall River, MA 02722-0869

Place this order form in an envelope with your  
Check or Money Order.



# Amaze Me

Please use this order form when subscribing to Amazing Computing™, ordering Back Issues, or ordering Amiga™ Public Domain Software

Name \_\_\_\_\_  
 Street \_\_\_\_\_  
 City \_\_\_\_\_ St. \_\_\_\_\_ Zip \_\_\_\_\_

**PROPER ADDRESS REQUIRED.** In order to expedite and guarantee your order, all large Public Domain Software orders, as well as most Back issue orders, are shipped by United Parcel Service. UPS requires that all packages be addressed to a street address for correct delivery.

Please circle the appropriate item: **New Subscription**

**Renewal**

Subscription: \$ \_\_\_\_\_

Please start my subscription to Amazing Computing™ with the next available issue or renew my current subscription. I have enclosed \$24.00 for 12 issues in the U.S. (\$36.00 Canada and Mexico, \$44.00 foreign surface). All funds must be in U.S. Currency on a U.S. Bank

Back Issues:

\$5.00 each US, \$6.00 each Canada and Mexico, \$7.00 each Foreign Surface.

Please circle your Back Issue choices below:

Vol1.1	Vol1.2	Vol1.3	Vol1.4	Vol1.5	Vol1.6	Vol1.7	Vol1.8	Vol1.9	Vol2.1	Vol2.2
Vol2.3	Vol2.4	Vol2.5	Vol2.6	Vol2.7	Vol2.8	Vol2.9	Vol2.10	Vol2.11	Vol2.12	Vol3.1
Vol3.2	Vol3.3	Vol3.4	Vol3.5	Vol3.6	Vol3.7	Vol3.8	Vol3.9	Vol3.10	Vol3.11	Vol3.12
Vol4.1										

Back Issues: \$ \_\_\_\_\_

**Public Domain Software:**

**\$6.00 each for subscribers (yes, even the new ones!)**

**\$7.00 each for non subscribers**

**(three disk minimum on all foreign orders)**

Please circle your Public Domain Software choices below:

**Amazing on Disk:**

A#1...Source & Listings V3.9

**Amicus:**

A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11	A12	A13	A14	A15	A16	A17
A18	A19	A20	A21	A22	A23	A24	A25	A26								

**Fred Fish:**

FF1	FF2	FF3	FF4	FF5	FF6	FF7	FF8	FF9	FF10	FF11	FF12	FF13	FF14	FF15	FF16	FF17
FF18	FF19	FF20	FF21	FF22	FF23	FF24	FF25	FF26	FF27	FF28	FF29	FF30	FF31	FF32	FF33	FF34
FF35	FF36	FF37	FF38	FF39	FF40	FF41	FF42	FF43	FF44	FF45	FF46	FF47	FF48	FF49	FF50	FF51
FF52	FF53	FF54	FF55	FF56	FFNA	FF58	FF59	FF60	FF61	FF62	FF63	FF64	FF65	FF66	FF67	FF68
FF69	FF70	FF71	FF72	FF73	FF74	FF75	FF76	FF77	FF78	FF79	FFNA	FF81	FF82	FF83	FF84	FF85
FF86	FF87	FFNA	FF89	FF90	FF91	FF92	FF93	FF94	FF95	FF96	FF97	FF98	FF99	FF100	FF101	FF102
FF103	FF104	FF105	FF106	FF107	FF108	FF109	FF110	FF111	FF112	FF113	FF114	FF115	FF116	FF117	FF118	FF119
FF120	FF121	FF122	FF123	FF124	FF125	FF126	FF127	FF128	FF129	FF130	FF131	FF132	FF133	FF134	FF135	FF136
FF137	FF138	FF139	FF140	FF141	FF142	FF143	FF144	FF145	FF146	FF147	FF148	FF149	FF150	FF151	FF152	FF153
FF154	FF155	FF156	FF157	FF158	FF159	FF160	FF161	FF162	FF163	FF164	FF165	FF166	FF167	FF168	FF169	FF170
FF171	FF172															

(NA Denotes disks removed from the collection)

PDS Disks: \$ \_\_\_\_\_

Please complete this form and mail with check or money order to:

**PIM Publications, Inc.**  
**P.O. Box 869**  
**Fall River, MA 02722-0869**

Total: \$ \_\_\_\_\_

Please allow 4 to 6 weeks for delivery



# DIGI-VIEW

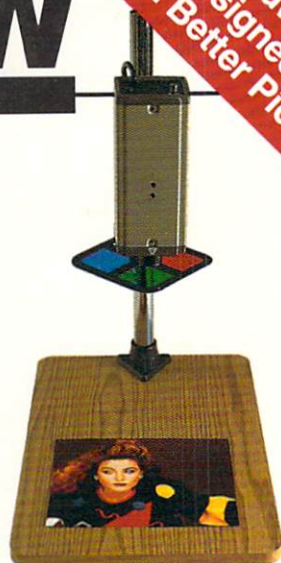
## GOLD

**All-New!**  
Hardware and Software  
Designed for A500/2000—  
Even Better Pictures Than Before!

1.



2.



Actual 4096 color Digi-View Gold picture

### 3. Simply the Best.

The all new Digi-View Gold is the best video digitizer for the Amiga. Period. Nothing else even comes close. Why? The secret is that Digi-View Gold captures 2.1 million colors in memory, giving you an incredible 100,000 apparent colors on screen simultaneously.

And it's easy to use. Just focus your video camera on any object or picture, and in seconds Digi-View Gold turns it into Amiga graphics that glow with vibrant color and clarity. Whether you are creating graphics for desktop publishing, presentations, video, or just for fun, Digi-View Gold gives you dazzling images with amazing simplicity.

Digi-View Gold is designed specifically for the Amiga 500 and 2000, and plugs directly into the parallel port. Digi-View Gold's powerful image capture and manipulation software (version 3.0) now has complete control of color and sharpness, full overscan, extra halfbrite, and a special line art mode for desktop publishing.

#### Only Digi-View Gold:

- Can digitize in all Amiga resolution modes from 320x200 up to 768x480 (full hi-res overscan)
- Uses 2 to 4096 colors (including extra halfbrite)
- Uses exclusive Enhanced HAM for super fine detail
- Is 100% IFF compatible and works with any graphics software
- Can digitize 21 bits per pixel (2.1 million colors) for the highest quality images possible
- Has advanced dithering routines that give an apparent 100,000 colors on screen simultaneously
- Has powerful Image processing controls for complete IFF picture manipulation

If you want the highest quality graphics for your Amiga, as easy as 1, 2, 3; then you need the new version of the best selling video digitizer of all time: Digi-View Gold.

**Only \$199.95**

Digi-View Gold is available now  
at your local Amiga dealer.  
Or call 1-800-843-8934

**NEWTEK**  
INCORPORATED

\*Requires standard gender changer for use with Amiga 1000. Video camera required; not included. NewTek sells a video camera, copy stand, and the Digi-Droid automated filter wheel for Digi-View Gold. If your local retailer doesn't carry these products, call us at 913-354-1146. Digi-View Gold is a trademark of NewTek, Inc. Amiga is a trademark of Commodore-Amiga, Inc. Be seeing you!!